

3100096007917

PERPUSTAKAAN
ITS

Tgl. Terima

22 SEP 1994

Terima Dari

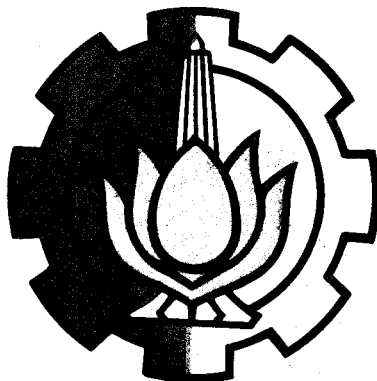
#

No. Agenda Prp.

2711

**KOTAK HITAM UNTUK MEREKAM KECEPATAN
LINEAR DAN RPM PADA KENDARAAN BERMOTOR
DENGAN MENGGUNAKAN
MIKROKONTROLER 68HC11E9**

RSE
621.391 6
Sur
L-1
1994



MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

Oleh :

Erma Suryani

NRP : 2882200981

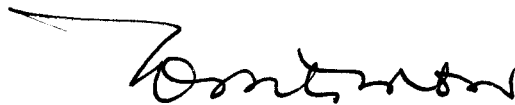
**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1994**

**KOTAK HITAM UNTUK MEREKAM KECEPATAN
LINEAR DAN RPM PADA KENDARAAN BERMOTOR
DENGAN MENGGUNAKAN
MIKROKONTROLER 68HC11E9**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar
Sarjana Teknik Elektro
Pada
Bidang Studi Teknik Elektronika
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya**

**Mengetahui / Menyetujui :
Dosen Pembimbing**



(Ir. NAWANTOWIBOWO H.)

**SURABAYA
AGUSTUS, 1994**

KATA PENGANTAR

Puji dan syukur kami panjatkan atas kehadiran Allah SWT atas karunia dan limpahan rahmat-Nya sehingga buku ini dapat terselesaikan pada waktunya.

Buku ini disusun sebagai pelengkap mata kuliah Tugas Akhir (EE1799) yang merupakan mata kuliah wajib guna memenuhi Satuan Kredit Semester pada jenjang pendidikan Strata I pada Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya. Adapun judul dari Tugas Akhir ini adalah :

KOTAK HITAM UNTUK MEREKAM KECEPATAN LINEAR DAN RPM PADA KENDARAAN BERMOTOR DENGAN MENGGUNAKAN MIKROKONTROLER 68HC11E9.

Atas terselesaikannya buku ini, terima kasih yang sebesar-besarnya penyusun sampaikan kepada :

1. Bapak Dr. Ir. Moh. Salehudin, M.Eng.Sc, selaku Ketua Jurusan Teknik Elektro ITS.
2. Bapak Ir. Soetikno selaku Koordinator Bidang Studi Elektronika Jurusan Teknik Elektro FTI - ITS.
3. Bapak Ir. Katjuk Astrowulan, MSEE.
4. Bapak Ir. Nawantowibowo sebagai Dosen Pembimbing dan Dosen Wali yang telah memberikan bimbingan selama penyelesaian Tugas Akhir ini.
4. Seluruh rekan-rekan Bidang Studi Elektronika .
6. Semua pihak yang telah membantu penyusun dalam menyelesaikan tugas akhir ini.

7. Bapak dan Ibu, yang tidak mungkin disebutkan lagi jasanya bagi penyusun serta seluruh keluarga yang tidak bosan-bosannya mendorong penyelesaian Tugas Akhir ini.

Bersama ini penyusun berharap agar buku ini dapat bermanfaat bagi rekan-rekan mahasiswa maupun para pembaca yang berminat.

Surabaya, Agustus 1994

Penyusun

ABSTRAK

Untuk meningkatkan kewaspadaan pengemudi kendaraan bermotor diperlukan suatu peralatan yang dapat mengambil data kecepatan linear (km/h) maupun kecepatan putaran mesin (rpm). Data-data ini diperlukan untuk memberikan peringatan batas kecepatan buat pengemudi.

Data-data kecepatan ini disimpan dalam RAM, dan selanjutnya dikirim ke IBM PC secara serial untuk diolah dan ditampilkan dalam bentuk grafik. Grafik ini meliputi kecepatan linear dan kecepatan putaran terhadap waktu. Dari grafik ini pemilik kendaraan (untuk kendaraan umum) dapat mengetahui apakah pengemudi sudah menjalankan kendaraannya sesuai dengan standard kecepatan dan rpm yang ada pada jenis kendaraannya.

DAFTAR ISI

JUDUL.....	i
HALAMAN PENGESAHAN.....	ii
KATA PENGANTAR.....	iii
ABSTRAK.....	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
I.1 LATAR BELAKANG	1
I.2 PERMASALAHAN	2
I.3 PEMBATASAN MASALAH.....	2
I.4 TUJUAN.....	3
I.5 RUANG LINGKUP	3
I.6 METODOLOGI.....	3
I.7 SISTEMATIKA PEMBAHASAN.....	3
I.8 RELEVANSI	4
BAB II TEORI PENUNJANG.....	5
II.1 PENDAHULUAN.....	5
II.2 KENDARAAN BERMOTOR.....	5

IL.2.1 MESIN (ENGINE).....	5
IL.2.2 CHASIS.....	7
IL.2.2.1 SISTEM PENERUS DAYA.....	8
IL.3 SISTEM KELISTRIKAN MOBIL.....	13
IL.3.1 PENGUKUR KECEPATAN (SPEEDOMETER).....	14
IL.3.1.1 MACAM-MACAM SPEEDOMETER.....	15
IL.3.2 PENGUKUR PUTARAN MESIN (RPM).....	17
IL.4 KECEPATAN LINIER.....	17
IL.5 KECEPATAN PUTARAN.....	19
IL.6 OPTOCOUPLER sebagai DETEKTOR KECEPATAN.....	20
IL.7 MIKROKONTROLER MC98H11E9.....	22
IL.7.1 DESKRIPSI PIN MC68HC11E9.....	25
IL.7.1.1 Power Supply V_{DD} dan Ground (V_{SS}).....	25
IL.7.1.2 Mode Select (MODB/VSTBY dan MODA/LIR).....	26
IL.7.1.3 Crystal driver (XTAL), Input Clock Eksternal.....	27
IL.7.1.4 Reset (RESET).....	27
IL.7.1.5 Interrupt Request (IRQ).....	27
IL.7.1.6 A/D Reference Voltages (V_{RL} , V_{RH}).....	28
IL.7.1.7 Strobe A dan Adress Strobe (STRA/AS).....	28
IL.7.1.8 Strobe B dan Read/Write (STRB/R/ W).....	28
IL.7.1.9 Sinyal Port.....	29

IL 7.2 MODE OPERASI MIKROKONTROLER MC68HC11E9.....	32
IL 7.2.1 SINGLE CHIP.....	32
IL 7.2.2 EXPANDED MULTIPLEXED.....	32
IL 7.2.3 SPECIAL BOOTSTRAP.....	33
IL 7.2.4 SPECIAL TEST.....	34
IL 7.3 MEMORY MAP MC68HC11E9.....	35
IL 7.4 PROGRAMABLE TIMER.....	36
IL 7.4.1 TIMER INPUT CAPTURE.....	37
IL 7.4.2 TIMER OUTPUT COMPARE.....	38
IL 7.5 SERIAL COMMUNICATION INTERFACE (SCI).....	40
IL 8 PPI 8255.....	46
IL 8.1 MODE 0 (BASIC I/O).....	46
IL 8.2 MODE 1 (STROBED I/O).....	46
IL 8.3 MODE 2 (STROBED BIDIRECTIONAL I/O).....	47
BAB III PERENCANAAN PERANGKAT KERAS.....	48
III.1 PENDAHULUAN.....	48
III.2 DIAGRAM BLOK HARDWARE.....	48
III.3 UNIT PERANGKAT KERAS.....	50
III.3.1 CENTRAL PROCESSING UNIT.....	50
III.3.2 RANGKAIAN BUFFER MIKROKONTROLER	
MC68HC11.....	54
III.3.3 MEMORY MAPPING.....	55

III.3.4 RANGKAIAN MEMORI EKSTERNAL.....	56
III.3.5 DETEKTOR KECEPATAN LINEAR (KM/H).....	58
III.3.6 DETEKTOR KECEPATAN PUTARAN (RPM).....	58
III.3.7 RANGKAIAN WARNING RPM dan DETEKSI POSISI GIGL.....	60
BAB IV PERENCANAAN PERANGKAT LUNAK.....	62
IV.1 PENDAHULUAN.....	62
IV.2 PERANGKAT LUNAK PADA MC68HC11E9.....	62
IV.2.1 ROUTINE-ROUTINE MIKROKONTROLER MC68HC11E9..	63
IV.2.1.1 ROUTINE INISIALISASI.....	63
IV.2.1.2 ROUTINE TIMER.....	65
IV.2.1.3 ROUTINE SERIAL.....	66
IV.2.1.4 ROUTINE INPUT/OUTPUT.....	67
IV.3 PERANGKAT LUNAK PADA KOMPUTER IBM PC.....	67
IV.3.1 KOMUNIKASI DATA SERIAL.....	67
IV.3.2 PENGOLAHAN DATA DAN TAMPILAN	68
BAB V PENGUJIAN DAN PENGUKURAN.....	69
V.1 PENGAMBILAN DATA.....	69
V.2 ANALISIS DATA	70
V.3 KETERANGAN GRAFIK.....	74
BAB VI KESIMPULAN DAN SARAN.....	75
VI.1 KESIMPULAN.....	75
VI.2 SARAN DAN KEMUNGKINAN PENGEMBANGAN	75

DAFTAR GAMBAR

Gambar 2.1 Motor Bolak-Balik.....	6
Gambar 2.2 Gas Turbine.....	7
Gambar 2.3 Prinsip Kerja Kopling.....	9
Gambar 2.4 Torsi.....	10
Gambar 2.5 Final Gear	12
Gambar 2.6 Prinsip Kerja Differensial.....	12
Gambar 2.7 Speedometer.....	15
Gambar 2.8 Konstruksi Speedometer Magnet.....	16
Gambar 2.9 Grafik X-t.....	18
Gambar 2.9a Pecahan I Grafik X-t.....	18
Gambar 2.9b Pecahan II Grafik X-t.....	19
Gambar 2.10 Posisi Benda Pada Gerak Melingkar.....	20
Gambar 2.11 Rangkaian Dasar Optocoupler.....	22
Gambar 2.12 Blok Diagram MC68HC11E9.....	23
Gambar 2.13 Register-Register pada Chip MC68HC11.....	24
Gambar 2.14 Konfigurasi Pin-Pin MC68HC11E9.....	25
Gambar 2.15 Demultiplexed untuk Address/Data	33
Gambar 2.16 Peta Memori MC68HC11E9.....	35
Gambar 2.17 Pengaturan Pembagi Clock Internal Timer.....	36
Gambar 2.18a Register TCNT	37
Gambar 2.18b Register TCTL2.....	37
Gambar 2.18c Register TFLG1.....	37
Gambar 2.18d Timer Input Capture.....	37

Gambar 2.19a Register TCNT.....	39
Gambar 2.19b Register TOC5.....	39
Gambar 2.19c Register TFLG1.....	39
Gambar 2.19d Register TMSK1.....	40
Gambar 2.19e Register TCTL1.....	40
Gambar 2.20 Blok Diagram Transmitter SCL.....	42
Gambar 2.21 Blok Diagram Receiver SCL.....	44
Gambar 2.22 Register-Register Kontrol SCL.....	45
Gambar 2.23 Mode Operasi PPI 8255.....	47
Gambar 3.1 Blok Diagram Hardware.....	49
Gambar 3.2 Diagram Waktu Mode Expanded.....	51
Gambar 3.3 Rangkaian Demultiplexer.....	52
Gambar 3.4 Decoder E dan R/W.....	53
Gambar 3.5 Rangkaian Buffer Data dan Alamat.....	54
Gambar 3.6 Perencanaan Memori mapping.....	55
Gambar 3.7 Rangkaian Memori Eksternal.....	57
Gambar 3.8 Rangkaian Pembentuk Pulsa.....	58
Gambar 3.9 Rangkaian Detektor Untuk Kecepatan Rpm.....	59
Gambar 3.10 Rangkaian Warning Rpm dan Deteksi Posisi Gigi.....	61
Gambar 4.1 Diagram Alir Program Mikrokontroler.....	64
Gambar 4.2 Inisialisasi TCTL2.....	65
Gambar 4.3 Inisialisasi Baud.....	65

DAFTAR TABEL

Tabel 2.1 Mode Operasi MCU.....	26
Tabel 2.2 Fungsi Port Tiap Bit.....	30
Tabel 2.3 Kondisi Bit HPRI0 untuk Tiap Mode Operasi.....	34
Tabel 3.1 Sinyal E dan R/W	53
Tabel 3.2 Decoding RAM dan EPROM.....	56
Tabel 3.2 Decoding Pengalamatan PPI 8255.....	60
Tabel 5.1 Tabel Pengambilan Data.....	69
Tabel 5.2 Tabel Data setelah Dikonversi.....	72

B A B I

PENDAHULUAN

I. 1 LATAR BELAKANG

Perkembangan penduduk dewasa ini menyebabkan banyak tuntutan. Salah satu tuntutan yang sangat vital adalah dalam hal transportasi. Perkembangan dalam bidang transportasi ini menyebabkan banyak timbul permasalahan, yaitu kecelakaan lalu lintas. Dari data yang ada menunjukkan bahwa kesalahan terbesar adalah dari faktor manusianya dalam hal ini pengemudi. Dan kendaraan yang sering mengalami kecelakaan adalah kendaraan umum. Untuk itu diperlukan suatu alat pemantau untuk merekam aktivitas yang dilakukan pengemudi kendaraan dan memberikan peringatan apabila melebihi batas kecepatan.

Peralatan ini akan dapat merekam beberapa parameter seperti kecepatan laju kendaraan dan kecepatan putaran mesin (rpm).

Selain pengambilan data alat ini juga dilengkapi dengan peringatan pada pengendara berupa bunyi jika kecepatannya melebihi batas yang diijinkan.

Dari data-data ini, maka pemilik kendaraan dapat mengetahui secara langsung bagaimana pengemudi menjalankan kendaraannya. Dari alat ini diharapkan dapat meningkatkan kewaspadaan pengendara sehingga keselamatan penumpang lebih terjamin.

I. 2 PERMASALAHAN

Dari uraian diatas maka perlu dibuat suatu alat yang dapat mencatat kecepatan kendaraan umum selama di perjalanan. Parameter-parameter yang dideteksi adalah sebagai berikut:

1. Kecepatan linear (km/jam).
2. Kecepatan putaran mesin (rpm).

Jika kecepatan putaran mesin (rpm) melebihi batas yang ditentukan buzzer akan berbunyi sebagai peringatan pada pengendara, untuk itu maka gigi persneling harus dinaikkan. Demikian juga jika kecepatan linear (km/h) melebihi batas maka buzzer juga akan berbunyi agar pengendara menurunkan kecepatannya.

Proses pengambilan data ini dilakukan secara bersamaan (multitasking). Setelah dilakukan proses pengambilan data, maka proses selanjutnya adalah memindahkan data dari RAM (Random Access Memory) ke komputer IBM PC melalui proses komunikasi serial diteruskan dengan proses pengolahan data dan menampilkan hasilnya dalam bentuk grafik.

Grafik-grafik ini meliputi grafik kecepatan linear (rpm) terhadap waktu, grafik kecepatan linear (km/jam) terhadap waktu.

I. 3 PEMBATAAN MASALAH

Peralatan yang dibuat pada Tugas akhir ini hanya dibatasi pada perekaman data kecepatan linier dan kecepatan putaran mesin. Dan banyaknya data yang dapat direkam digunakan memory sebesar 24 KiloByte. Namun demikian dari dua

parameter yang diperoleh akan dapat dievaluasi bagaimana pengemudi menjalankan kendaraannya dengan baik.

I. 4 TUJUAN

- ♦ Meningkatkan kewaspadaan pengemudi
- ♦ Menerapkan teknologi mikrokontroler MC68HC11 dalam pembuatan kotak hitam untuk kendaraan umum.

I. 5 RUANG LINGKUP

Alat ini dapat digunakan untuk mengambil data dari berbagai parameter dan kemudian dikomunikasikan secara serial ke IBM PC untuk diolah dan ditampilkan dalam bentuk grafik.

I. 6 METODOLOGI

Untuk mengerjakan tugas akhir ini digunakan metode sebagai berikut:

1. Studi literatur tentang MC68HC11.
2. Perencanaan hardware.
3. Perencanaan software.
4. Pembuatan alat dan uji coba.
5. Menyusun laporan Tugas Akhir.

I. 7 SISTEMATIKA PEMBAHASAN

Bab I merupakan pendahuluan yang memuat latar belakang, permasalahan, tujuan, ruang lingkup, metodologi, sistematika pembahasan dan relevansi.

Bab II berisi tentang teori penunjang yang meliputi sistem kerja kendaraan bermotor, sistem kerja optocoupler sebagai detektor kecepatan dan mikrokontroler MC68HC11.

Bab III berisi perencanaan perangkat keras (hardware) yang akan dibuat.

Bab IV berisi perencanaan perangkat lunak (software) yang menunjang perangkat keras yang dibuat.

Bab V berisi pengujian dan pengukuran yang meliputi pengambilan data frekwensi yang dikonversi menjadi kecepatan.

Bab VI merupakan penutup dari seluruh materi tugas akhir ini yang berisi kesimpulan dan saran untuk menyempurnakan sistem peralatan yang telah dibuat.

I. 8 RELEVANSI

Peralatan dirancang sedemikian rupa sehingga dapat diandalkan untuk perjalanan jauh sehingga alat ini dapat berfungsi sebagai kontrol bagi pengendara agar lebih berhati-hati jika menjalankan kendaraan untuk lebih menjamin keselamatan penumpang dan bagi si pemilik kendaraan dapat menghemat biaya operasional karena peralatannya tidak akan cepat aus.

B A B II

TEORI PENUNJANG

II.1 PENDAHULUAN

Pada bab ini akan diuraikan teori penunjang yang digunakan dalam membahas rancangan peralatan kotak hitam untuk kendaraan umum dengan menggunakan mikrokontroler MC68HC11E9.

II. 2 KENDARAAN BERMOTOR

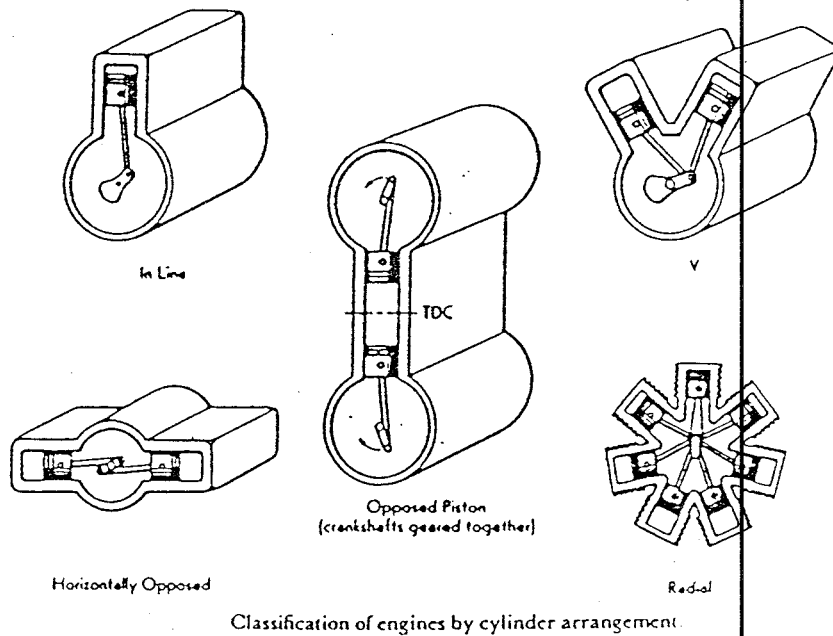
Secara umum kendaraan bermotor terdiri dari tiga komponen utama yang fungsinya berbeda dan bekerja secara bersama-sama. Komponen utama yang membentuk sebuah kendaraan bermotor yaitu mesin, chasis dan sistem kelistrikan. Fungsi ketiga komponen tersebut adalah :

1. Mesin (Engine) untuk menghasilkan tenaga gerak.
2. Chasis untuk tempat kedudukan mesin, mentransmisikan tenaga gerak mesin menjadi gerakan kendaraan, dan mengendalikan arah kendaraan.
3. Sistem kelistrikan untuk mengatur kerja mesin melalui sistem pengapian, sistem pengatur bahan bakar, assesoris.

II. 2. 1 MESIN (ENGINE)

Berdasarkan gerakan mekanisme pengubah energi dapat dibedakan menjadi:

a. Motor Bolak-Balik (Reciprocating Engine)



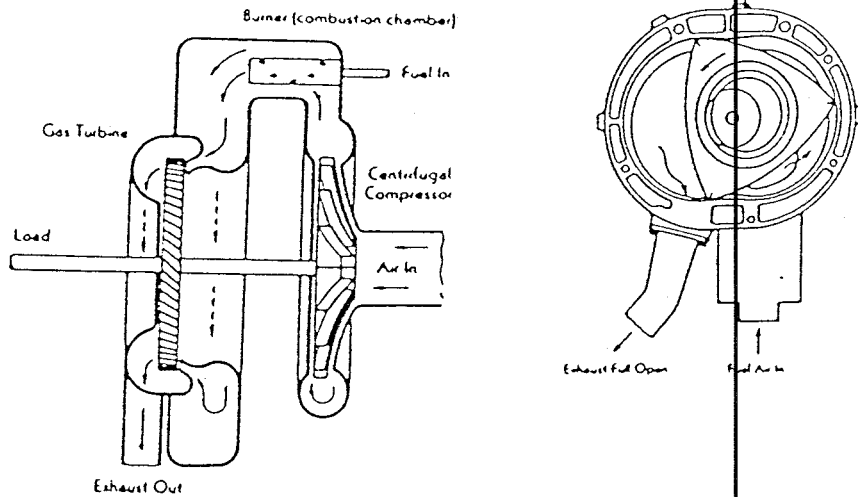
Gambar 2.1¹
Motor Bolak-Balik

Energi hasil pembakaran diubah menjadi energi gerak melalui gerakan bolak-balik dari piston. Gerakan bolak-balik ini kemudian diubah menjadi gerak putar oleh poros engkol (crank shaft).

b. Motor Rotary (Rotary Engine)

Motor ini dapat memanfaatkan energi hasil pembakaran menjadi gerak putar secara langsung sehingga gerakannya menjadi lebih halus dan efisien.

¹



Gambar 2.2²
Gas Turbine

II. 2. 2 CHASSIS

Chassis merupakan tempat komponen-komponen penting sebuah mobil seperti:

- ♦ Mesin.
- ♦ Sistem penerus daya.
- ♦ Poros dan Suspensi.
- ♦ Sistem kemudi.
- ♦ Roda dan ban.
- ♦ Sistem rem.
- ♦ Pengapian.

- ♦ Sistem kelistrikan, dll.

Dibawah ini akan diuraikan beberapa komponen penunjang diantaranya :

II. 2. 2. 1 SISTEM PENERUS DAYA (POWER TRAIN SYSTEM)

Sistem penerus daya meliputi kopling, transmisi, dan gardan/differential.

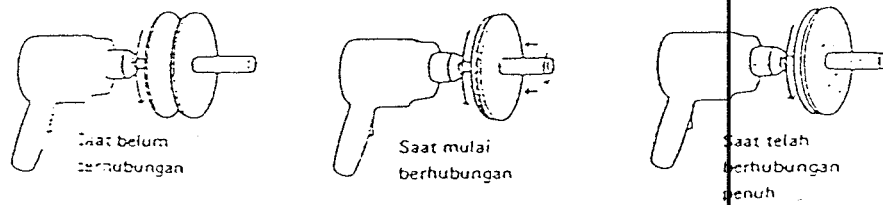
A. KOPLING

Berfungsi untuk memutuskan dan menghubungkan putaran mesin ke transmisi (pemisah beban). Pada saat mobil berhenti akan tersentak bila tenaga mesin tiba-tiba menjadi besar dan juga mesin akan mati mendadak bila tenaga kecil tetapi bebannya berat. Oleh karena itu diperlukan pemindahan tenaga sedikit demi sedikit.

Prinsip kerja kopling

Prinsip kerja kopling dapat dianalogikan seperti kerja bor tangan yang dihubungkan dengan piringan seperti ditunjukkan pada gambar 2.3

Bor diumpamakan sebagai mesin dan piringan sebagai roda gila. Sebuah piringan lain yang bebas didekatkan pada bor. Pada saat kedua piringan belum bersentuhan maka piringan kedua tidak akan berputar. Pada saat bersentuhan dengan tekanan lemah maka piringan kedua akan turut berputar dengan kecepatan yang lebih rendah dari piringan pertama. Sedangkan pada saat kedua piringan bersentuhan dengan tekanan yang kuat maka kedua piringan akan berputar dengan kecepatan yang sama.

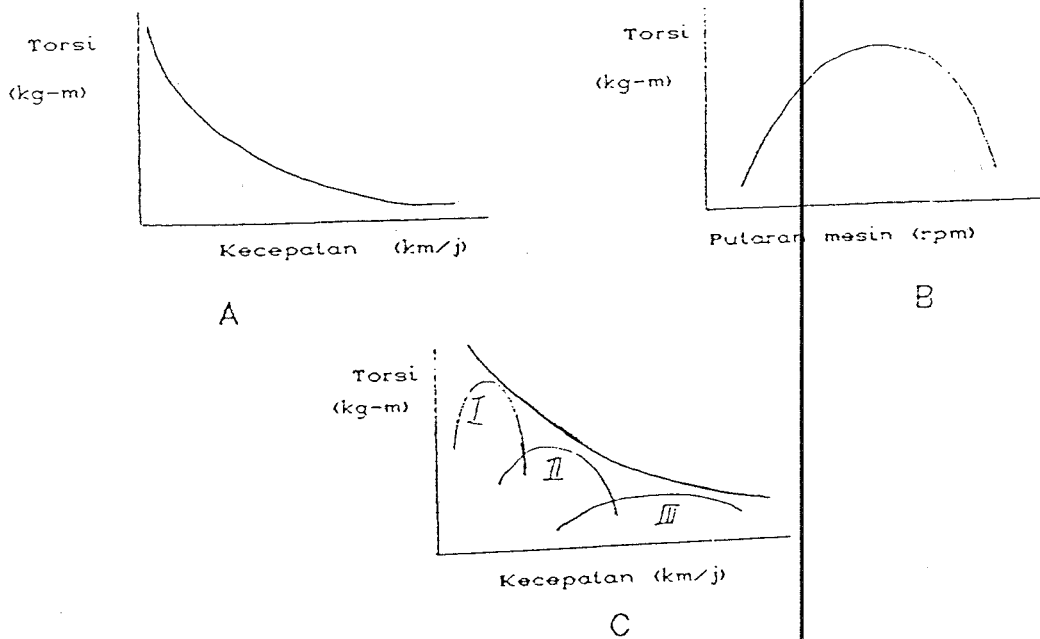


Gambar 2.3³
Prinsip Kerja Kopling

B. TRANSMISI

Putaran yang dihasilkan oleh mesin sebelum mencapai harus disesuaikan dengan kebutuhan terlebih dahulu. Pada saat kendaraan start atau mendukung beban yang berat maka kendaraan membutuhkan torsi yang besar. Sebaliknya jika kendaraan melaju di jalan bebas dengan kecepatan tinggi maka yang dibutuhkan bukan torsi yang besar melainkan putaran yang tinggi pada rodanya. Putaran tinggi dan torsi yang besar bersumber dari mesin. Sedang besarnya torsi maksimum mesin tidaklah sebesar yang dibutuhkan kendaraan. Oleh karena itu dibutuhkan suatu sistem yang mampu memperbesar torsi dengan jalan memperkecil putaran mesin (misalnya pada saat start dan mengembalikan putaran mesin bila kendaraan sudah berjalan. Gambar 2.4 menunjukkan grafik torsi(kg-m) terhadap kecepatan, putaran mesin, dan kecepatan setelah diberikan transmisi.

Untuk memenuhi kebutuhan tersebut digunakan sistem transmisi. Transmisi memvariasikan kecepatan putaran mesin yang sampai ke roda dengan menggunakan perbandingan gigi sehingga didapat karakteristik yang hampir sama dengan yang dibutuhkan kendaraan. Fungsi sistem transmisi:

Gambar 2.4⁴

- a. Torsi yang dibutuhkan kendaraan.
- b. Torsi yang dihasilkan mesin.
- c. Torsi yang didapat setelah melalui sistem transmisi.

- ♦ mengatur putaran mesin ke roda.
- ♦ memperbesar momen.
- ♦ mempertinggi kecepatan.
- ♦ mengubah arah putaran mesin.

Dengan sistem transmisi ini maka kecepatan putar mesin akan diturunkan secara bertingkat untuk memperhalus perpindahannya.

C. GARDAN/DIFFRENSIAL

Pada saat mobil berbelok roda-roda penggerak (roda-roda depan untuk front wheel drive dan roda-roda belakang untuk rare wheel drive) pada bagian kanan dan kiri akan mendapat beban yang berbeda dan menjalani lintasan yang berbeda jauhnya dalam waktu yang sama. Karena itu keduanya harus mempunyai kecepatan yang berbeda agar salah satu roda tidak terseret yang dapat menyebabkan patahnya poros.

Perbedaan putaran diatur oleh suatu sistem pemindah daya yang disebut differensial yang terdiri dari dua bagi bagian utama yaitu:

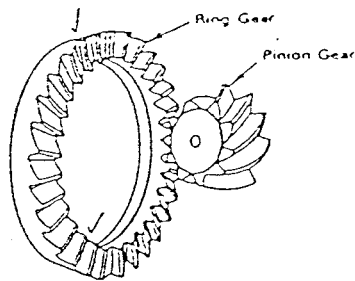
- ♦ Final gear.
- ♦ Differensial gear.

C. 1 FINAL GEAR

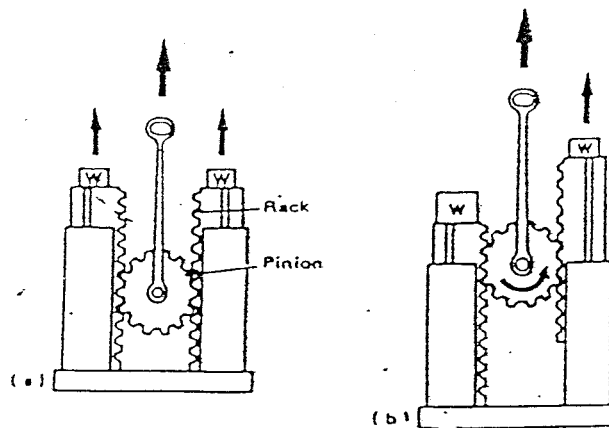
Final gear ditunjukkan pada gambar 2.5. Final gear ini berfungsi untuk memperbesar momen dan mengubah arah putaran. Momen yang dihasilkan oleh transmisi pada saat-saat tertentu tidak cukup untuk menggerakkan kendaraan sehingga putaran harus diturunkan lagi dengan final gear. Selain itu jika tidak digunakan final gear maka gigi tranmisi harus dibuat dalam dimensi besar dan mempunyai kekuatan yang besar. Hal ini tentunya tidak ekonomis.

C. 2 DIFFERENSIAL GEAR

Prinsip kerja differensial gear ditunjukkan dalam gambar 2.6



Gambar 2.5⁵
Final Gear



Gambar 2.6⁶
Prinsip kerja differensial

Pada gambar 2.6a ditunjukkan suatu sistem terdiri atas rak dan pinion. Kedua rak mempunyai beban (gaya) yang sama besar. Dalam kondisi ini apabila pinion ditarik ke atas maka kedua rak akan tertarik ke atas dengan ketinggian yang sama. Pinion hanya akan

bergerak ke atas dan tidak berputar pada porosnya. Di sini ditunjukkan bahwa tidak terjadi perbedaan tinggi rak kiri dan rak kanan apabila beban sama.

Pada gambar 2.6b beban pada rak sebelah kiri lebih besar dari sebelah kanan. Akibatnya pada saat pinion ditarik ke atas pinion akan berputar pada porosnya dan rak sebelah kanan akan terangkat lebih tinggi dari rak sebelah kiri. Disini ditunjukkan bahwa apabila kedua beban pada rak tidak sama besar dan pinion ditarik ke atas maka terlihat pinion berputar pada porosnya sekaligus mengangkat rak kiri dan kanan tidak sama tinggi.

II. 3 SISTEM KELISTRIKAN MOBIL

Sistem kelistrikan mobil adalah bagian yang penting karena sistem ini merupakan sumber tenaga penggerak. Secara umum sistem kelistrikan pada mobil dapat dibedakan menjadi:

1. Sistem kelistrikan bahan bakar:
 - a. L-Jetronic.
 - b. K-Jetronic.
2. Sistem pengapian:
 - a. Kondensor.
 - b. Koil
- c. Distributor.
- d. Kabel-kabel tegangan tinggi.
- e. Tahanan balast.

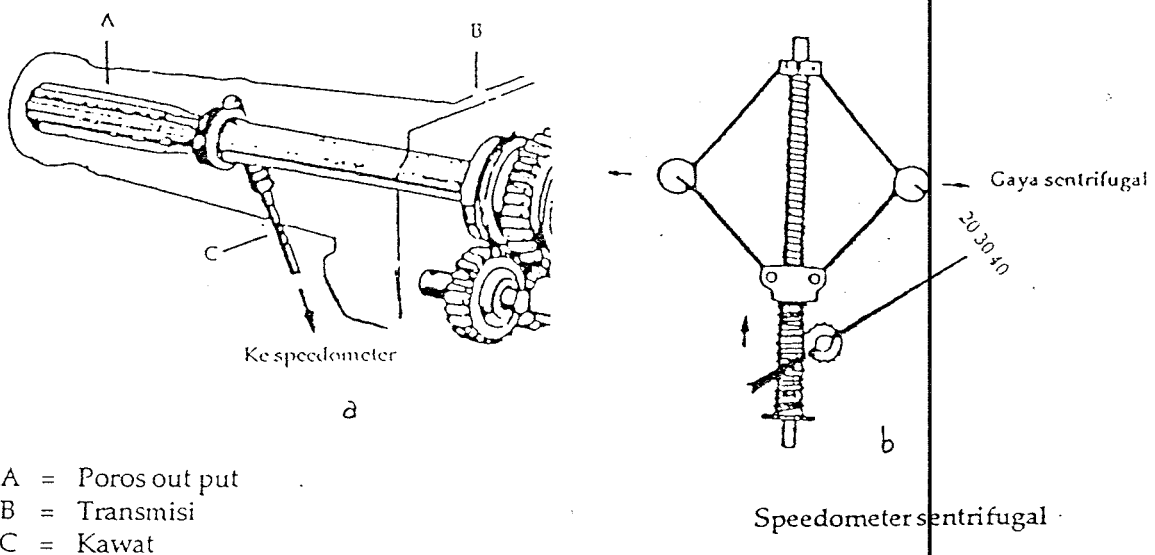
3. Sistem Pengukuran:
 - a. Pengukuran bahan bakar.
 - b. Pengukuran tekanan oli.
 - c. Pengukur suhu air pendingin.
 - d. Speedometer dan odometer.
4. Sistem lampu dan alat bantu
 - a. Lampu-lampu penerangan.
 - b. Lampu tanda.
 - c. Penghapus kaca.
5. Sistem pengisian:
 - a. Dinamo.
 - b. Alternator.
 - c. Regulator.
 - d. Dioda.
 - e. Baterai.

II. 3. 1 PENGUKUR KECEPATAN (SPEEDOMETER)

Alat ukur ini menunjukkan kecepatan kendaraan yang biasanya dikombinasikan dengan odometer untuk mencapai jarak perjalanan yang ditempuh.

Speedometer digerakkan oleh gigi yang dipasang pada ujung belakang poros output transmisi. Gerakan ini diteruskan melalui kawat fleksibel. Satuan kecepatan adalah km/jam.

Odometer mencatat jarak tempuh kendaraan sejak keluar dari pabrik, dan mempunyai kemampuan mencatat sampai 99.999,9 km. Adapula beberapa kendaraan yang dilengkapi dengan dua odometer. Salah satu odometer hanya mencatat dengan kapasitas kecil dimana penunjukan akan lebih cepat kembali ke angka nol. Odometer jarak pendek ini biasanya disebut trip-meter. Gambar speedometer dapat dilihat pada gambar 2.7



Gambar 2.7

- a. Speedometer yang Digerakkan oleh Transmisi
b. Speedometer Sentrifugal

II. 3. 1. 1 MACAM-MACAM SPEEDOMETER

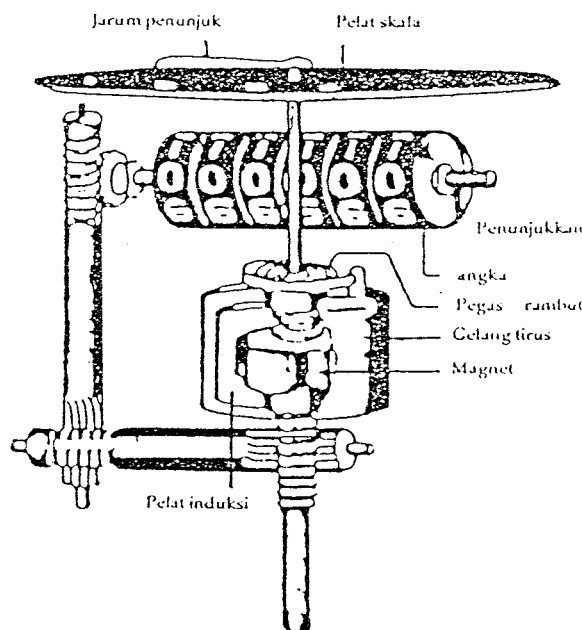
Dewasa ini dikenal beberapa jenis speedometer seperti speedometer sentrifugal, speedometer magnit dan speedometer listrik. Yang paling banyak digunakan adalah speedometer magnit.

1. Speedometer sentrifugal

Seperti terlihat pada gambar 2.7b jenis ini memakai gaya sentrifugal yang ditimbulkan oleh dua buah bola bobot. Jika poros berputar, bola-bola akan berputar mengarah ke luar dan tabung bergigi akan terangkat, kemudian memutar ponion dan penunjuk akan bergerak menunjukkan kecepatan kendaraan.

2. Speedometer magnet

Konstruksi jenis ini terlihat pada gambar 2.8. Disini digunakan gaya tarik dari magnet permanen. Magnet ini berbentuk roda diputar oleh kawat fleksibel. Pelat yang menerima induksi medan magnet dihubungkan dengan jarum penunjuk melalui sebuah poros. Bila roda



Gambar 2.8^a
Konstruksi Speedometer Magnet

magnet berputar, maka akan timbul perputaran garis gaya sehingga akan terbentuk momen yang dapat menarik pelat induksi. Momen ini akan semakin besar bila kendaraan bergerak lebih cepat dan angka yang ditunjukpun semakin besar pula. Agar supaya gerakan jarum penunjuk stabil maka momen ini diimbangi dengan pegas rambut. Dengan demikian jika berhenti, pegas rambut ini akan mengembalikan jarum penunjuk ke angka nol.

II. 3. 2 PENGUKUR PUTARAN MESIN (RPM)

Alat ukur ini menunjukkan banyaknya putaran mesin tiap menit. Rpm ini digerakkan oleh gigi-gigi yang dipasang pada ujung belakang poros output transmisi.

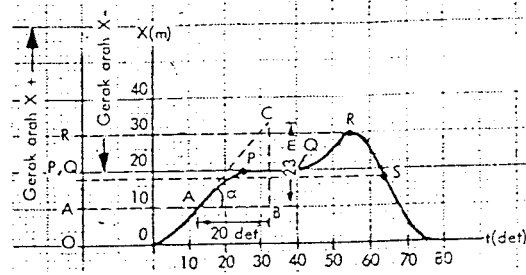
Gerakan bermula dari putaran mesin kemudian melalui kawat fleksibel diteruskan ke alat ukur.

Konstruksi rpm adalah sama dengan speedometer hanya saja pada rpm tanpa diberi odometer, sehingga konstruksinya lebih sederhana.

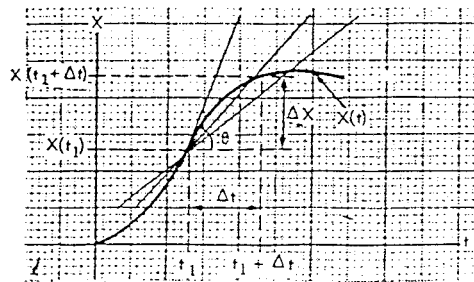
II. 4 KECEPATAN LINEAR

Untuk mengetahui kecepatan pada suatu saat maka kita dapat menggunakan deskripsi grafik seperti ditunjukkan dalam gambar 2.9. Grafik ini diperoleh dari pengamatan gerak.

Untuk lebih jelasnya bagian grafik disekitar saat t_1 dapat dilukiskan lagi seperti dalam gambar 2.9a. Dari grafik ini dapat dilihat bahwa:



Gambar 2.9⁹
Grafik X-t



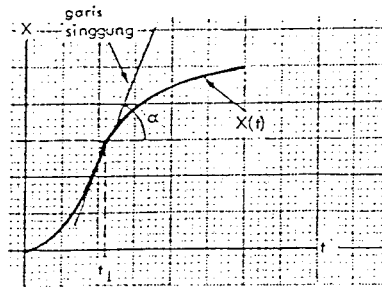
Gambar 2.9a
Pecahan 1 Grafik X-t

$$V(t_1) = \lim_{\delta t \rightarrow 0} \frac{X(t_1 + \delta t) - X(t_1)}{\delta t}$$

$$= \lim_{\delta t \rightarrow 0} \text{tg } \theta$$

Dari grafik 2.9b dapat dilihat bahwa $V(t_1) = \operatorname{tg} \alpha$, yaitu koefisien garis singgung $X(t)$ pada saat $t = t_1$ atau

$$V(t_1) = \frac{dx}{dt} \Big|_{t=t_1} = \operatorname{tg} \alpha$$



Gambar 2.9b
Pecahan II Grafik $X - t$

Untuk lebih membantu maka dibuat segitiga ABC siku-siku sembarang sehingga didapat $\operatorname{tg} \alpha = \frac{BC}{AB} = \frac{x}{t}$

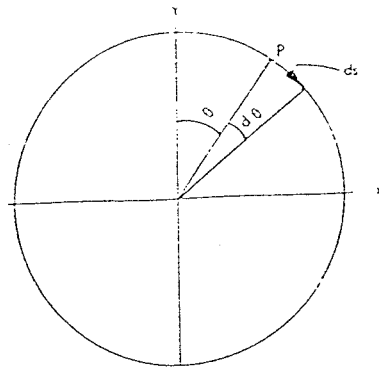
II.5 KECEPATAN PUTARAN

Untuk mengukur kecepatan putaran kita dapat menganalisa dari gerak melingkar. Dalam gerakan jarak benda terhadap titik pusat lingkaran adalah tetap dan sama dengan jari-jari lingkaran sehingga posisi benda terhadap titik pusat lingkaran cukup dinyatakan dengan sudut θ seperti ditunjukkan dalam gambar 2.10.

Panjang busur ds dapat dinyatakan dengan $ds = r d\theta$ sehingga didapat:

$$V = \frac{dS}{dt} = r \frac{d\theta}{dt}$$

dimana: $\frac{d\theta}{dt} = \text{kecepatan sudut} = \omega$.



Gambar 2.10¹⁰
Posisi Benda Pada Gerak Melingkar

Satuan kecepatan sudut dinyatakan dalam rad/detik. Persamaan diatas menjadi :

$$V = r w$$

dimana : w = kecepatan sudut

r = jari-jari lingkaran.

Waktu yang diperlukan dalam gerak melingkar untuk menempuh satu putaran disebut periode putaran. Besaran lain yang sering digunakan adalah frekwensi yang menyatakan berapa kali partikel mengelilingi lingkaran dalam satuan waktu. Seringkali frekwensi dinyatakan dalam rpm (revolution/minute) atau putaran per menit.

II. 6 OPTOCOUPLER SEBAGAI DETEKTOR KECEPATAN

Sebagai detektor kecepatan digunakan optocoupler. Dengan menggunakan optocoupler dan piringan berlubang yang terpasang pada sumbu motor, maka akan diperoleh pulsa setiap terjadi pemotongan sinar pada basis optocoupler oleh sirip yang ikut berputar. Dengan menghitung selang waktu antara pulsa pertama dan pulsa berikutnya maka

¹⁰ Ibid, Hal. 22.

kecepatan dapat dihitung. Besarnya kecepatan berbanding terbalik dengan periode dan berbanding lurus dengan konstanta tertentu.

Optocoupler merupakan suatu komponen yang terdiri dari Light Emitter, Light Path dan Photo-Transistor yang dikemas dalam suatu kemasan. Optocoupler digunakan untuk mengisolasi electric antara suatu rangkaian listrik dengan rangkaian listrik lainnya agar tidak terjadi umpan balik.

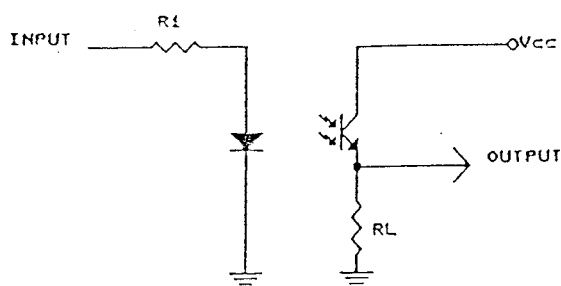
Prinsip kerja dari optocoupler ini dapat dijelaskan sebagai berikut :

- ♦ Sinyal yang masuk pada kaki masukan optocoupler diubah kedalam bentuk cahaya oleh LED infra merah. Kuat lemahnya cahaya yang dihasilkan bergantung dari besar sinyal masukan yang diterima oleh LED tersebut.
- ♦ Pancaran akan diterima oleh phototransistor dan dengan menambahkan resistor pull-up pada ujung kolektor berubah-ubah.

Rangkaian dasar Optocoupler dapat dilihat pada gambar 2.11.

Optocoupler mempunyai beberapa kelebihan antara lain:

1. Adanya isolasi antara rangkaian internal dan eksternal.
2. Switch dengan kecepatan tinggi.
3. Tahan terhadap goncangan atau getaran mekanik cukup tinggi.
4. Ukuran dan bentuknya kecil dan sederhana, sehingga mempermudah dalam pemakaian dan efisiensi tempat.



Gambar 2.11¹¹
Rangkaian Dasar Optocoupler

II.7 MIKROKONTROLER MC98H11E9

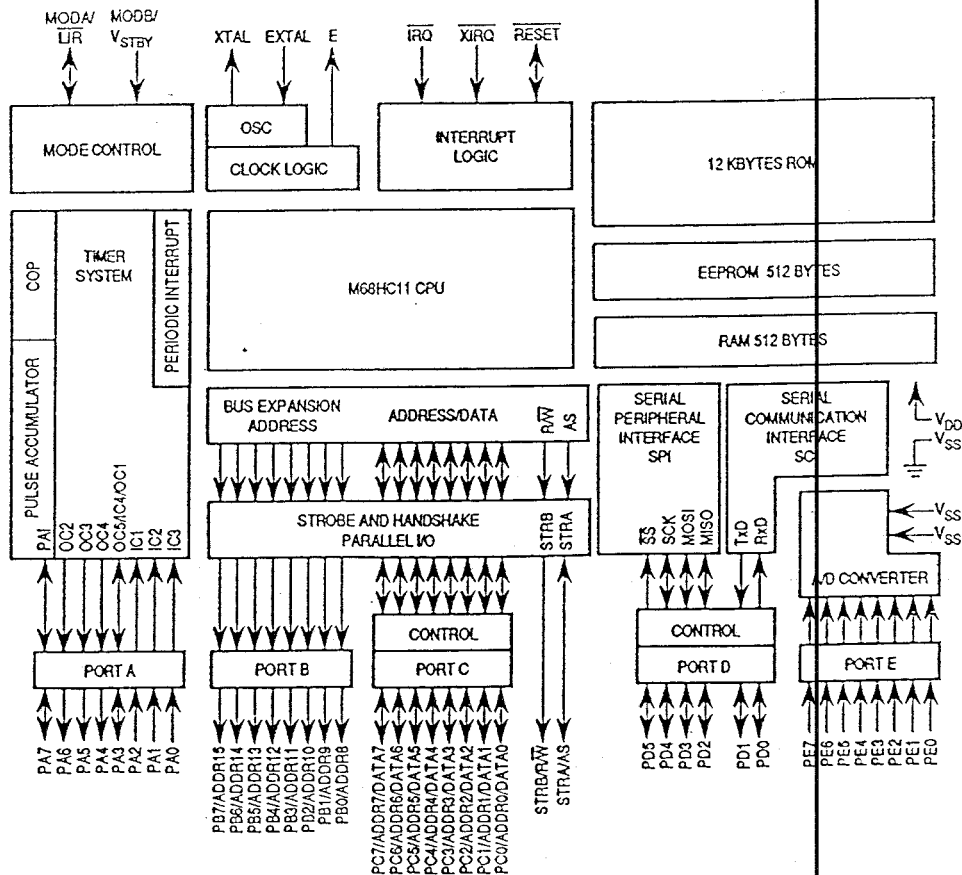
Mikrokontroler MC68HC11E9 adalah salah satu anggota keluarga mikrokontroler MC68HC11. Mikrokontroler ini dikemas dalam sebuah chip dengan teknologi High Density CMOS (HCMOS) sehingga mempunyai kecepatan akses bus sampai 3 MHz dengan konsumsi daya yang rendah.

Dalam chip ini terdapat beberapa fasilitas antara lain:

- ♦ ROM sebesar 12 Kilo byte.
- ♦ EEPROM (Electrical Erasable Programable Read Only Memory) sebesar 512 byte.
- ♦ RAM sebesar 512 byte.
- ♦ Sistem Timer 16 bit
- ♦ Rangkaian pulsa akumulator 8 bit.
- ♦ Rangkaian Real Time Interrupt .
- ♦ Sistem Watchdog.
- ♦ Interface peralatan serial (SPI).

- ♦ Interface komunikasi serial (SCI).
- ♦ 8 channel, 8 bit Analog to Digital Converter.

Dalam chip ini juga terdapat rangkaian self monitoring untuk mencegah sistem error dan sistem clock monitor yang dapat membangkitkan sistem reset jika clock external hilang atau terlalu lambat. Blok diagram dari chip MC68HC11E9 terdapat dalam gambar 2.12.

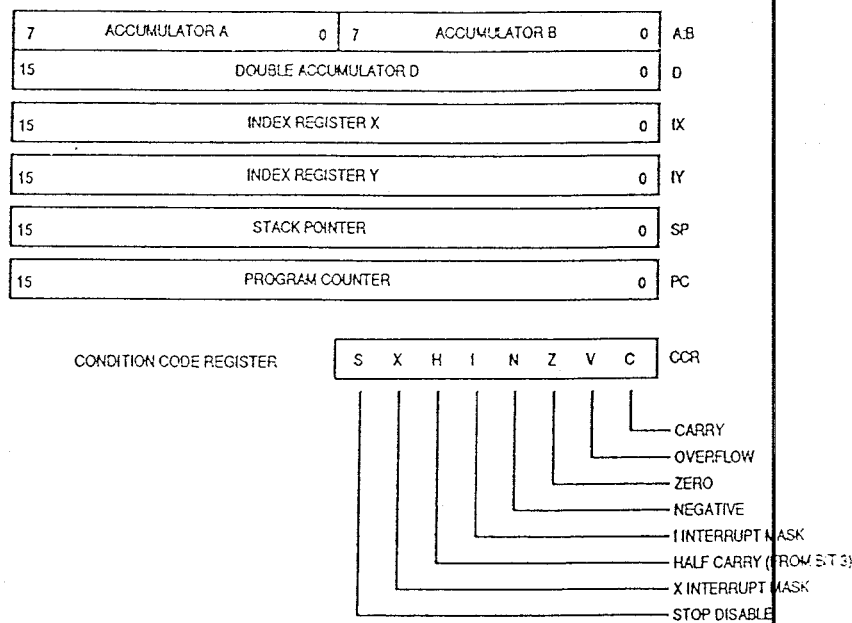


Gambar 2.12²
Blok Diagram MC68HC11E9

Terdapat tujuh buah register CPU untuk pemrograman antara lain :

- ♦ 8 bit akumulator A dan B .
- ♦ Double akumulator D.
- ♦ Register index X.
- ♦ Register index Y.
- ♦ Stack pointer.
- ♦ Program Counter.
- ♦ Register condition code.

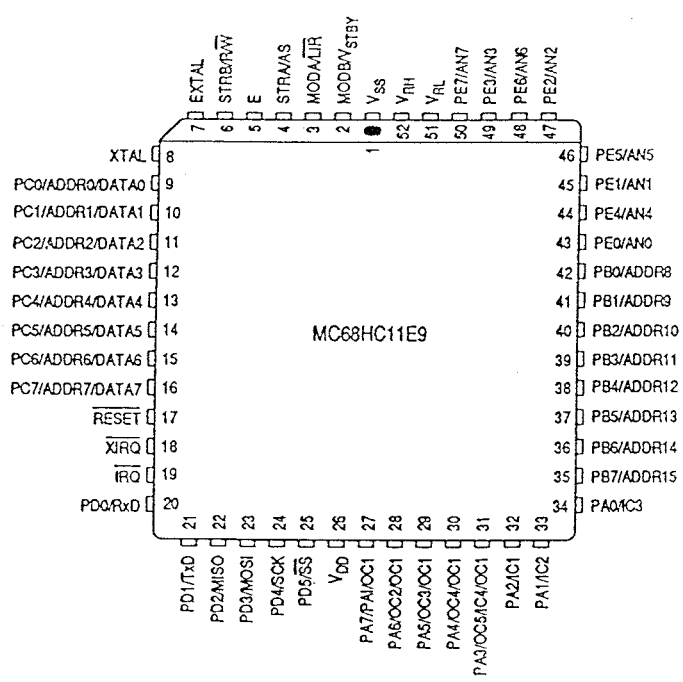
Gambar tujuh buah register ini dapat dilihat pada gambar 2.13.



Gambar 2.13⁵
Register-Register pada Chip MC68HC11E9

II. 7. 1 DESKRIPSI PIN MC68HC11E9

Chip MC68HC11E9 ini terdiri dari 52 pin yang dikemas secara quad pack seperti terdapat dalam gambar 2.14.



Gambar 2.14¹⁴
Konfigurasi Pin-Pin MC68HC11E9

II. 7. 1. 1 POWER SUPPLY V_{DD} DAN GROUND (V_{SS})

V_{DD} adalah input supply daya positif dan V_{SS} adalah ground. Pin V_{DD} dan V_{SS} ini digunakan untuk supply daya pada mikrokontroler.

14 Ibid, Hal. 2-1.

IL 7. 1. 2 MODE SELECT (MODB/VSTBY DAN MODA/LIR)

Pin Mode B/VSTBY (standby RAM) ini mempunyai fungsi ganda yaitu sebagai input mode select dan sebagai power supply standby. Mode A/ Load Instruction register (MODA/LIR) digunakan untuk memilih mode operasi selama MCU reset, dan berfungsi sebagai sinyal diagnostic output (output open drain) selama MCU mengeksekusi instruksi.

Tabel 2.1 menunjukkan kondisi bit pin-pin untuk masing-masing mode operasi.

Tabel 2.1¹⁵
Mode Operasi MCU

MODE A	MODE B	MODE SELECTED
1	0	Single Chip
1	1	Expanded Multiplexed
0	0	Special Bootstrap
0	1	Special Test

LIR menjadi low selama siklus E clock yang pertama untuk setiap instruksi yang dikerjakan. Sinyal VSTBY digunakan sebagai input tegangan supply RAM sesaat. Jika tegangan pada pin ini lebih dari tegangan ambang MOS (0,7 volt) di atas tegangan V_{DD} , RAM dalam MCU akan disupply oleh pin ini, sehingga isi RAM tetap dapat dipertahankan tanpa adanya V_{DD} .

¹⁵ Ibid., Hal. 4-8.

II. 7. 1. 3 CRYSTAL DRIVER (XTAL), INPUT CLOCK EKSTERNAL (EXTAL) DAN E CLOCK OUTPUT

E clock adalah clock output frekwensi bus yang dapat digunakan sebagai sinyal referensi basic timing. E clock low saat terjadi proses internal dan high jika data masuk ke mikrokontroler.

Pin XTAL dan EXTAL dihubungkan ke sebuah kristal atau sebuah rangkaian oscillator CMOS untuk mengontrol rangkaian pembangkit clock internal. Frekwensi clocknya sebesar empat kali frekwensi yang dikeluarkan oleh E clock output. Jika digunakan sebuah oscillator dari CMOS maka output oscillator tersebut dihubungkan ke pin EXTAL, sedang pin XTAL dihubungkan ke resistor 10k sampai 100k ke ground untuk mengurangi noise yang timbul.

II. 7. 1. 4 RESET

Pin ini digunakan untuk kontrol sinyal secara bidirectional yaitu sebagai input atau output. Sebagai input (aktif low) pin ini digunakan untuk menginisialisasi kondisi awal /start up MC68HC11E9. Sebagai output pin ini berfungsi untuk menunjukkan suatu kegagalan dalam mikrokontroler yang dideteksi oleh rangkaian watchdog.

II. 7. 1. 5 INTERRUPT REQUEST (IRQ)

Pin ini berfungsi untuk mendeteksi adanya permintaan interrupt asinkron ke MCU. Dengan memanfaatkan OPTION register melalui program yang dibuat maka pin ini dapat dipilih untuk sinyal interruptnya harus negative edge sensitive triggering atau level sensitif

triggering. Bila digunakan untuk mereset maka pin ini dikonfigurasi level sensitif triggering.

Pin ini dihubungkan dengan resistor pullup 4,7k ke V_{DD} .

II. 7. 1. 6 A/D REFERENCE VOLTAGES (V_{RL}/V_{RH})

Dua pin ini digunakan sebagai input tegangan referensi ADC di dalam MCU. V_{RL} untuk tegangan referensi rendah yaitu 0 Vdc. V_{RH} untuk tegangan referensi tinggi minimum 3 volt dan maksimum V_{DD} .

II. 7. 1. 7 STROBE A DAN ADRESS STROBE (STRA/AS)

Pin ini memiliki fungsi yang terpisah sifatnya tergantung mode operasi yang dipilih. Pada mode single chip, STRA berfungsi sebagai strobe input yang dapat diprogram untuk handshaking peralatan I/O luar yang dihubungkan ke MCU secara paralel. Pada mode expanded multiplexed pin AS berfungsi sebagai output digunakan untuk demultiplexing alamat dan data pada port C.

II. 7. 1. 8 STROBE B DAN READ/WRITE (STRB/R/ W)

Pin ini digunakan untuk mengatur arah data bus tergantung dari mode operasi yang digunakan. Untuk mode operasi single chip output STRB dipakai sebagai programable strobe untuk sinyal handshaking pada peralatan I/O yang dihubungkan ke MCU secara paralel. Pada mode operasi expanded multiplex pin R/W digunakan untuk mengontrol arah transfer data dari data bus. Jika pin ini berada dalam kondisi rendah berarti data sedang dikirim/ditulis ke bus data. Jika R/W berada dalam kondisi tinggi berarti data bus sedang dibaca oleh MCU.

IL 7.1.9 SINYAL PORT

Mikrokontroler MC68HC11E9 mempunyai lima buah port yang masing-masing terdiri dari 8 bit. Port-port ini meliputi port A, port B, port C, port D dan port E. Port A, D, dan E tidak tergantung dari mode operasi, sedang port B dan port C dipengaruhi mode operasi.

Pada operasi single chip port B berfungsi sebagai output dan port C berfungsi sebagai input/output.

Jika dipilih mode expanded multiplexed port B merupakan address line yang berkondisi tinggi dan port C digunakan untuk multiplex address dan data.

Tabel 2.2 menunjukkan secara fungsi tiap pin dari port mikrokontroler untuk setiap mode operasi.

A. PORT A

Port A mempunyai fungsi yang sama untuk setiap mode operasi. Port ini digunakan sebagai empat input capture dan empat output compare atau tiga input capture dengan lima output compare, atau tiga input capture, satu input pulsa akumulator dan empat output compare. Setiap pin pada port A yang tidak difungsikan sebagai timer dapat dipakai sebagai general purpose I/O.

B. PORT B

Pada operasi single chip, semua pin pada port B berfungsi untuk general purpose output. Port B juga dapat digunakan untuk strobe output mode dimana pulsa output akan muncul pada sinyal STRB setiap saat data dikeluarkan (ditulis) ke port B.

Tabel 2.2¹⁶
Fungsi Port Tiap Bit

Port/Bit	Single-Chip and Bootstrap Mode	Expanded Multiplexed and Special Test Mode
PA0	PA0/IC3 PA1/IC2 PA2/IC1 PA3/OC5/IC4/and-or OC1 PA4/OC4/and-or OC1 PA5/OC3/and-or OC1 PA6/OC2/and-or OC1 PA7/PA1/and-or OC1	
PA1		
PA2		
PA3		
PA4		
PA5		
PA6		
PA7		
PB0	PB0	ADDR8
PB1	PB1	ADDR9
PB2	PB2	ADDR10
PB3	PB3	ADDR11
PB4	PB4	ADDR12
PB5	PB5	ADDR13
PB6	PB6	ADDR14
PB7	PB7	ADDR15
PC0	PC0	ADDR0/DATA0
PC1	PC1	ADDR1/DATA1
PC2	PC2	ADDR2/DATA2
PC3	PC3	ADDR3/DATA3
PC4	PC4	ADDR4/DATA4
PC5	PC5	ADDR5/DATA5
PC6	PC6	ADDR6/DATA6
PC7	PC7	ADDR7/DATA7
PD0	PD0/RxD PD1/TxD PD2/MISO PD3/MOSI PD4/SCK PD5/SS	
PD1		
PD2		
PD3		
PD4		
PD5		
—	STRA	AS
—	STRB	R/W
PE0	PE0/AN0 PE1/AN1 PE3/AN2 PE3/AN3 PE4/AN4 PE5/AN5 PE6/AN6 PE7/AN7	
PE1		
PE2		
PE3		
PE4		
PE5		
PE6		
PE7		

Pada mode operasi expanded multiplexed, semua pin port B berfungsi sebagai sinyal output untuk address. Selama siklus MCU, bit 8 sampai 15 dari address bus dikeluarkan melalui PB0 - PB7.

C. PORT C

Pada mode operasi single chip, semua pin port C berfungsi sebagai general purpose I/O. Data input pada port C dapat disimpan sementara dalam register dengan memberikan input transisi ke sinyal STRA. Port C dapat digunakan untuk full handshake paralel I/O dimana input STRA dan output STRB digunakan untuk control handshaking.

D. PORT D

Pin PD0-PD5, dapat digunakan sebagai sinyal input/output. Pin-pin ini juga dapat digunakan sebagai komunikasi serial interface (SCI) dan komunikasi peralatan serial (SPI).

Pin PD0 adalah sinyal input data yang diterima (RXD) untuk SCI, pin PD1 adalah sinyal data yang dikeluarkan (TXD) untuk SCI, PD2-PD5 digunakan untuk SPI. PD2 adalah sinyal Master In Slave Out (MOSI). PD3 adalah sinyal Master Out Slave In (MOSI). PD4 adalah sinyal clock serial (SCK) dan PD5 digunakan untuk sinyal slave (SS).

E. PORT E

Port E digunakan untuk input analog to digital conversion. Pembacaan data pada port E selama proses sampling data untuk konversi analog ke digital dapat menyebabkan ketidak validan data hasil konversi.

II. 7. 2 MODE OPERASI MIKROKONTROLER MC68HC11E9

Terdapat empat mode operasi yang digunakan untuk MC68HC11E9 antara lain :

- ♦ single chip
- ♦ expanded multiplexed
- ♦ special bootstrap
- ♦ special test

Mode operasi yang sering digunakan adalah mode operasi single chip dan expanded multiplexed. Dalam mode single chip hanya menggunakan memori di dalam chip MC68HC11E9. Sedangkan pada mode operasi expanded multiplexed dapat menggunakan memori dari luar selain memori dalam chip sendiri.

Mode Special bootstrap merupakan variasi dari mode single chip, yaitu mode khusus yang mempergunakan bootloader program dari ROM bootstrap. Mode special test merupakan mode khusus yang digunakan oleh pabrik pembuat chip untuk mentes chip MC68HC11E9 ini.

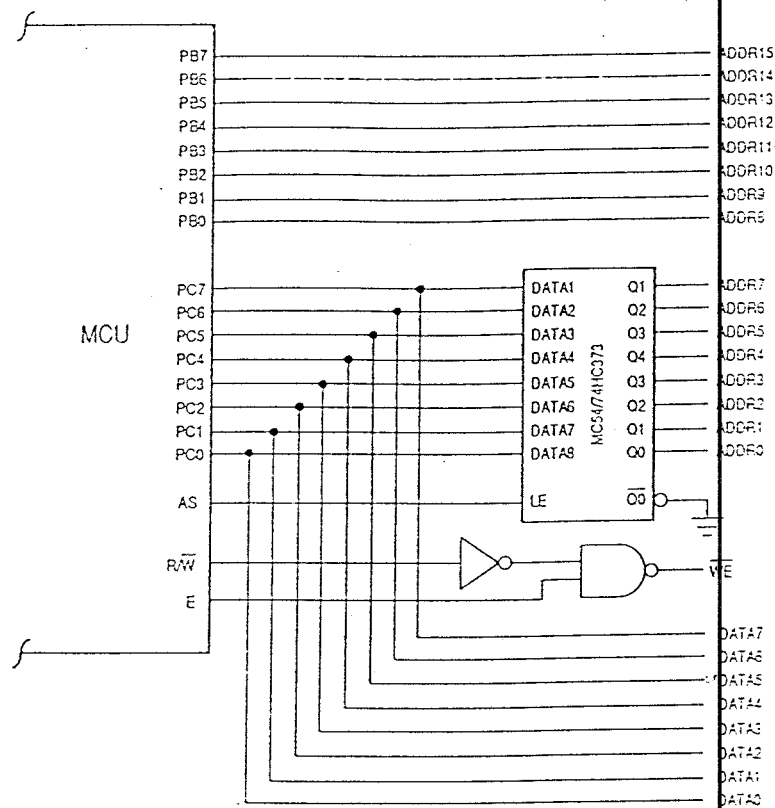
II.7.2.1 SINGLE CHIP

Pada operasi ini, MC68HC11E9 berfungsi sebagai mikrokontroler tanpa address atau data bus dari luar. Port B, port C, strobe A dan strobe B berfungsi sebagai general purpose input/output dan sinyal handshake.

II. 7. 2. 2 EXPANDED MULTIPLEXED

Pada mode operasi ini, MC68HC11E9 mempunyai kemampuan untuk mengakses 64K byte ruang address. Jumlah ini adalah sama dengan address memori pada chip yang digunakan untuk mode operasi single chip ditambah dengan address untuk peralatan dan

memori dari luar. Bus ekspansi dibuat pada port B dan C serta sinyal kontrol AS dan R/W. Gambar 2.15 menunjukkan cara menambah bus ekspansi untuk pengalamatan device atau memori dari luar melalui port B dan port C.



Gambar 2.15¹⁷
Demultiplexing untuk Address/Data

II. 7. 2. 3 SPECIAL BOOTSTRAP

Dalam mode ini MCU setelah reset akan mengaktifkan address \$BF00-\$BFFF yang berisi program bootloader yang termuat dalam ROM bootstrap.

II. 7. 2. 4 SPECIAL TEST

Mode ini khusus digunakan untuk mentes chip MC68HC11E9. Jadi hanya digunakan oleh pabrik pembuat chip ini. Pada mode ini reset dan interrupt vektor diset pada address \$BFFF-\$BFC0.

Keempat mode operasi yang telah dijelaskan di atas dapat dipilih melalui pin MODA dan MODB. Kondisi bit kedua pin ini menentukan jenis mode operasi yang dipilih dan juga menentukan kondisi bit special mode (SMOD) dan mode A (MDA) dalam register HPRIO. Kondisi bit yang dipengaruhi oleh mode operasi yang dipilih ini adalah untuk bit RBOOT, SMOD, MDA, dan IRVNE. Untuk lebih jelasnya hal ini dapat dilihat dalam tabel 2.3.

Tabel 2.3¹⁸
Kondisi Bit HPRIO untuk Tiap Mode Operasi

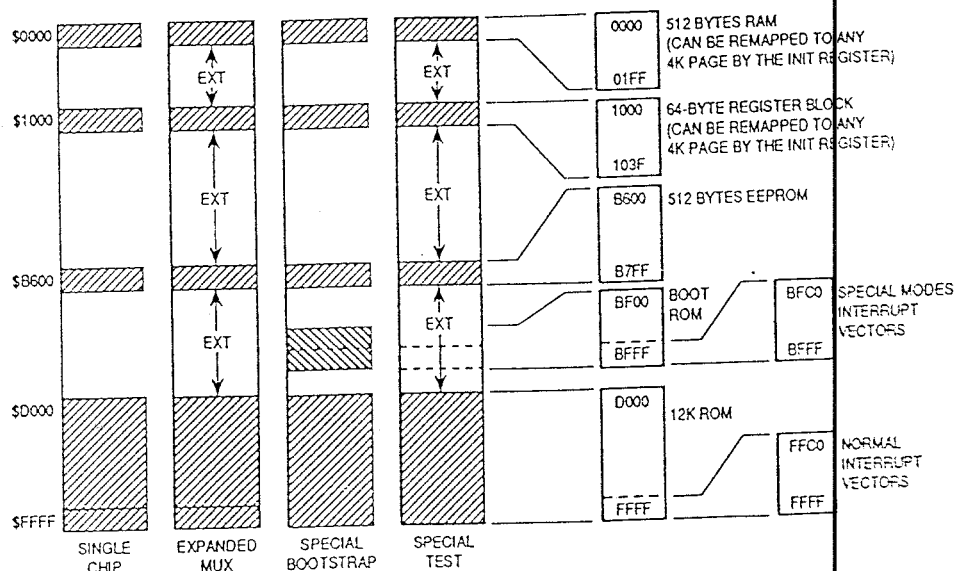
Inputs		Mode	Control Bits In HPRIO		
MODB	MODA		RBOOT	SMOD	MDA
1	0	Single Chip	0	0	0
1	1	Expanded Multiplexed	0	0	1
0	0	Special Bootstrap	1	1	0
0	1	Special Test	0	1	1

Kondisi Bit control RBOOT akan menentukan apakah bootloader ROM aktif pada address \$BF00-\$BFFF atau tidak. Kondisi bit SMOD akan mempengaruhi apakah MCU

pada mode normal atau mode special. Kondisi bit MDA akan menentukan apakah MCU menggunakan mode eksternal atau tidak. Kondisi bit IRVNE akan mempengaruhi apakah MCU dapat melakukan pembacaan data pada bus eksternal atau tidak.

II. 7.3 MEMORY MAP MC68HC11E9

Gambar 2.16 menunjukkan memori map untuk masing-masing mode operasi. Dari gambar dapat dilihat bahwa memori map untuk mode single chip dan expanded multiplexed hampir sama, hanya untuk single chip tidak menggunakan eksternal memori. Sedang pada expanded multiplexed digunakan memori eksternal yang besarnya sampai 64Kbyte. Jika terdapat alamat yang sama antara RAM eksternal dan memori internal serta register internal, maka digunakan urutan prioritas hardware sebagai berikut : register, RAM dan ROM.



Gambar 2.16¹⁹
Peta Memori MC68HC11E9

RAM internal 512 byte adalah RAM statis yang bersifat menyimpan data, variabel dan instruksi sementara. ROM bootstrap 256 byte adalah ROM internal yang memuat program bootloader yang akan muncul pada address \$BF00-\$BFFF bila MCU berada pada mode special bootstrap. 12 Kbyte ROM berada pada lokasi \$D000-\$FFFF untuk semua mode. 512 byte EEPROM berada pada lokasi \$B600-\$B7FF dan mempunyai waktu siklus baca yang sama dengan internal ROM. EEPROM dapat diprogram dan dihapus isi memorinya oleh software.

IL 7.4 PROGRAMABLE TIMER

Rangkaian timer MC68HC11E9 didukung oleh sebuah clock tunggal 2-Mhz jika digunakan kristal eksternal 8-MHz. Clock ini selanjutnya dapat dibagi lagi dengan pengaturan bit 0 dan 1 dari register TMSK2 (Timer Interrupt Mask Register 2) seperti terlihat dalam gambar 2.17. Pengaturan ini harus dilakukan sebelum 64 siklus pertama setelah reset.

Bit 7	6	5	4	3	2	1	0	\$1024
TOI	RTI	PAOVI	PAI	0	0	PR1	PRO	

PR0	PR1	Divide 2-Mhz internal clock by
0	0	1
0	1	4 → Writeable only during
1	0	8 64 cycles after reset
1	1	16

Gambar 2.17*
Pengaturan Pembagi Clock Internal untuk Timer

II. 7. 4. 1 TIMER INPUT CAPTURE

\$100E \$100E, \$100F

Bit 15	14	13	12	11	10	9	Bit 8
Bit 7	6	5	4	3	2	1	Bit 0

\$100F

Gambar 2.18a²¹
Register TCNT

Bit 7 6 5 4 3 2 1 Bit 0 \$1021

DG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A
------	-------	-------	-------	-------	-------	-------	-------

Gambar 2. 18b²²
Register TCTL2

Bit 7 6 5 4 3 2 1 Bit 0 \$1020

OC1F	OC2F	OC3F	OC4F	I4/OC5F	IC1F	IC2F	IC3F
------	------	------	------	---------	------	------	------

Gambar 2. 18c²³
Register TFLG1

TIC1–TIC3 — Timer Input Capture

\$1010–\$1015

\$1010	Bit 15	14	13	12	11	10	9	Bit 8	TIC1 (High)
\$1011	Bit 7	6	5	4	3	2	1	Bit 0	TIC1 (Low)
\$1012	Bit 15	14	13	12	11	10	9	Bit 8	TIC2 (High)
\$1013	Bit 7	6	5	4	3	2	1	Bit 0	TIC2 (Low)
\$1014	Bit 15	14	13	12	11	10	9	Bit 8	TIC3 (High)
\$1015	Bit 7	6	5	4	3	2	1	Bit 0	TIC3 (Low)

RESET: Input capture registers not affected by reset

Gambar 2. 18d
Timer Input Capture

21 Op.Cit., Hal. 9-10.
22 Op.Cit., Hal. 9-6.
23 Op.Cit., Hal. 9-12.

Tiap timer Input Capture dilengkapi dengan register 16 bit sebagai penyimpan nilai counter jika diberikan sinyal pada IC_i ($i = 1, 2, 3$). Sinyal tersebut dapat berupa transisi naik, transisi turun, atau setiap transisi, baik naik maupun turun, yaitu dengan pengaturan bit 0 sampai 5 pada register TCTL2 (Timer Control Register 2) seperti terdapat dalam gambar 2.18b.

Interrupt akan dienable jika bit IC_i pada register TMSK1 diset 1. Sebaliknya interrupt akan didisable. Sedangkan flag untuk masing-masing interrupt terdapat pada register TFLG1. Flag tersebut akan diset 1 jika terjadi interrupt. Untuk mereset kembali dapat dilakukan dengan menulis bit 1 ke bit bersesuaian yang akan direset.

II. 7. 4. 2 TIMER OUTPUT COMPARE

Register-register yang berkaitan dengan timer output compare 2, 3, 4, dan 5 diperlihatkan pada gambar 2.19a, 2.19b, 2.19c, dan 2.19d. Masing-masing timer dilengkapi dengan register 16 bit yang berisi nilai counter pada saat OC_i ($i=2, 3, 4, 5$) diinginkan berubah ke kondisi tertentu. Jadi kondisi output OC_i akan berubah jika nilai counter pada register TCNT sama dengan nilai pada register TOC_i. Nilai pada TOC_i dapat diubah dengan penulisan langsung ke register tersebut.

Kondisi tersebut dapat berupa kondisi tinggi, rendah atau toggle yang ditentukan oleh register TCTL1 (Timer Control Register 1) seperti pada gambar 2.19e. Disamping itu dapat pula diset agar output OC_i tidak dipengaruhi oleh nilai counter. Hal terakhir terjadi jika hanya diharapkan terjadinya interrupt tanpa disertai perubahan pada OC_i.

Mask bagi interrupt dikontrol oleh register TMSK1 (Timer Interrupt Mask

Register 1) seperti diperlihatkan pada gambar 2.19d. Jika bit yang bersesuaian diset 1, maka interrupt di-enable. Sebaliknya, jika bit yang bersesuaian direset 0, maka interrupt di-disable.

Setiap terjadi interrupt, maka bit flag yang bersesuaian pada register TFLG1 (Timer Interrupt Flag Register 1) akan diset 1. Flag tersebut dapat direset kembali dengan menuliskan bit 1 ke bit flag tersebut.

TCNT — Timer Counter

\$100E, \$100F

\$100E	Bit 15	14	13	12	11	10	9	Bit 8	TCNT (High)
\$100F	Bit 7	6	5	4	3	2	1	Bit 0	TCNT (Low)

Gambar 2.19a
Register TCNT

TI4/O5 — Timer Input Capture 4/Output Compare 5

\$101E, \$101F

\$101E	Bit 15	14	13	12	11	10	9	Bit 8	TI4/O5 (High)
\$101F	Bit 7	6	5	4	3	2	1	Bit 0	TI4/O5 (Low)

RESET: Forces TI4/O5 to \$FFFF

Gambar 2.19b
Register TOC5

TFLG1 — Timer Interrupt Flag 1

\$1023

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F
RESET:	0	0	0	0	0	0	0	0

Gambar 2.19c
Register TFLG1

TMSK1 — Timer Interrupt Mask 1

\$1022

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I
RESET:	0	0	0	0	0	0	0	0

Gambar 2.19d
Register TMSK1

TCTL1 — Timer Control 1

\$1020

	Bit 7	6	5	4	3	2	1	Bit 0
	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5
RESET:	0	0	0	0	0	0	0	0

Gambar 2.19e²⁴
Register TCTL1

MC68HC11E9 mempunyai sistem timer 16 bit free running counter, yang memperoleh clock output dari four stage prescaler (dibagi dengan 1,4,8,16) yang didrive oleh E clock MCU. Counter mulai menghitung dari \$0000-\$FFFF . Setelah counter mencapai maksimum, counter akan kembali lagi ke \$0000 dan overflow flag diset. Fungsi input disebut input capture yang digunakan untuk mencatat hitungan free running counter untuk mendeteksi edge pada input line. Fungsi output disebut output compare, sebab output akan beraksi ketika terjadi match antara 16 bit output register dengan free running counter.

II. 7. 5 SERIAL COMMUNICATION INTERFACE (SCI)

SCI untuk MC68HC11E9 adalah jenis universal asynchronous receiver transmitter (UART) yang didalamnya terdapat format NRZ (satu start bit, delapan atau sembilan data

24 Op.Cit., Hal. 9-10

bit dan satu stop bit) dengan berbagai variasi baud rate. Fungsi transmitter dan receiver SCI tidak saling tergantung, tetapi menggunakan data format dan bit rate yang sama.

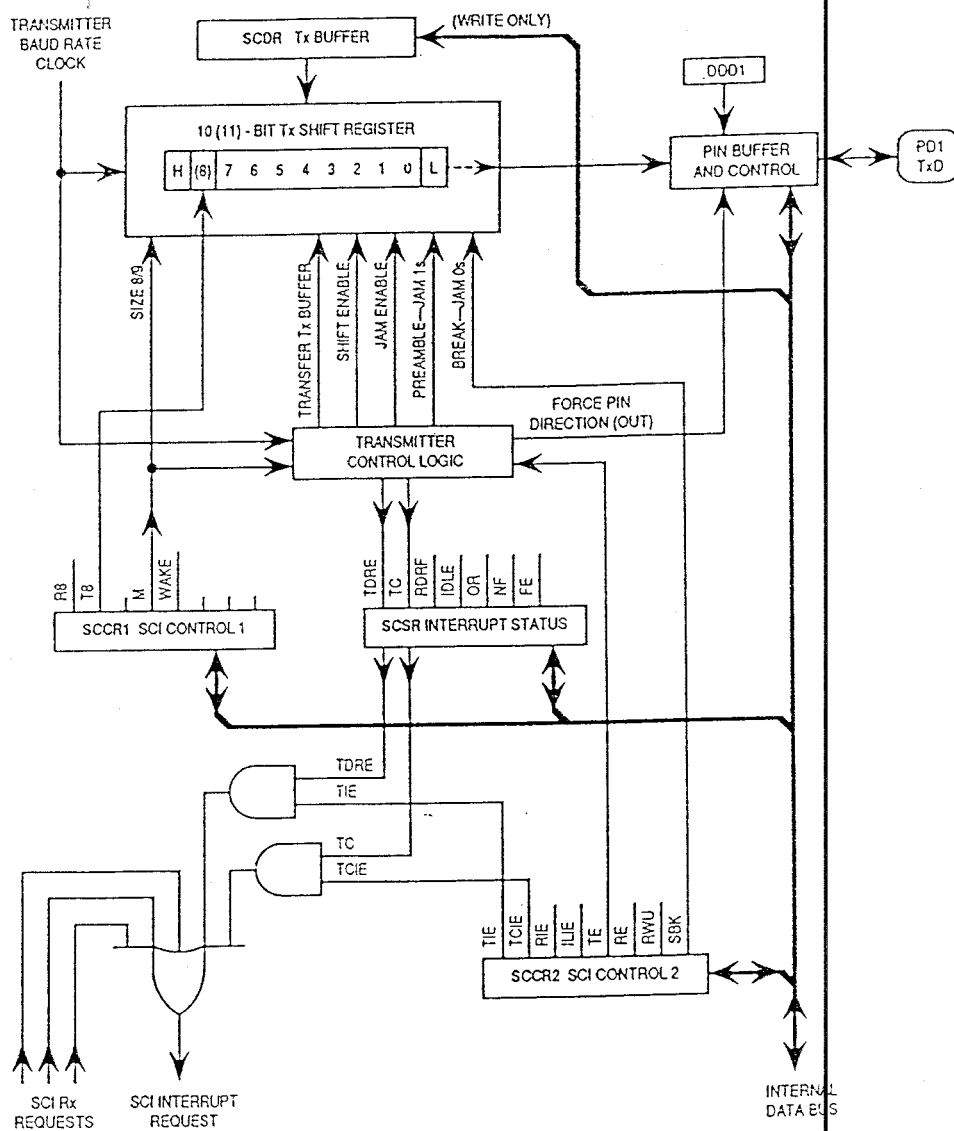
Beberapa kondisi yang dibutuhkan untuk format data serial :

1. Kondisi idle line adalah kondisi logika 1 sebelum pengiriman atau penerimaan data atau karakter.
2. Start bit, kondisi logika 0, menunjukkan awal data atau karakter yang dikirim atau diterima.
3. Bit data pertama yang dikirim/diterima adalah least significant byte (LSB).
4. Stop bit, kondisi logika 1, menunjukkan akhir dari suatu frame data.
5. Kondisi break didefinisikan sebagai pengiriman atau penerimaan data berkondisi logika 0 sedikitnya sepanjang satu frame.

Transmitter SCI dapat menghasilkan sinyal idle line dan break. Blok diagram transmitter SCI terdapat dalam gambar 2.20.

Jantung transmitter SCI adalah pada serial transmit shift register yang mengambil data dari transmit buffer. Data akan masuk ke dalam buffer apabila software yang dibuat melakukan penulisan data ke SCI data register (SCDR). Ketika data ditransfer dari buffer shift register, start bit berkondisi logika 0 yang pertama ditransfer kemudian baru delapan atau sembilan bit data, dan terakhir adalah stop bit. Satu frame data akan dikeluarkan ke TxD jika bit TE pada serial communication control register 2 (SCCR2) pada kondisi logika 1. Bit T8 pada SCI control register 1 (SCCR1) berlaku sebagai bit ke-9 pada format

data atau karakter. Bit ini digunakan hanya jika bit M pada SCCR1 pada kondisi logika 1 yang berarti data yang dikirim atau diterima adalah 9 bit. Bit TDRE dan TC secara



Gambar 2.20²⁵
Blok Diagram Transmitter SCI pada MC68HC11E9

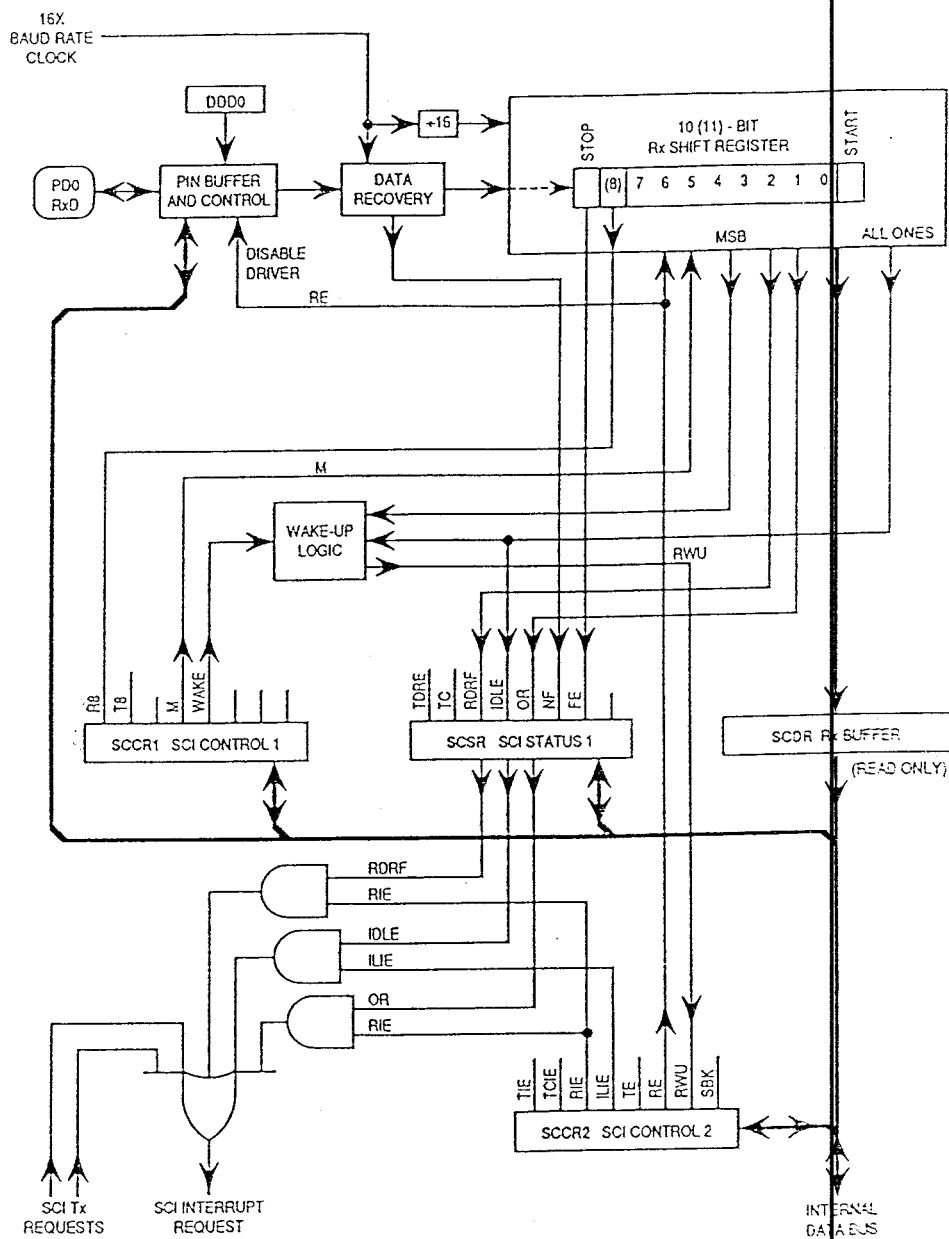
otomatis diset oleh transmitter control logic. Kedua bit ini dapat dibaca melalui software. Bit-bit transmit interrupt enable (TIE), transmit complete interrupt enable (TCIE), TDRE dan TC digunakan untuk membangkitkan sinyal SCI interrupt request.

Pada operasi penerimaan data, maka operasinya adalah kebalikan dari operasi pengiriman data. Blok diagram receiver SCI terdapat dalam gambar 2.21.

Jantung dari receiver SCI adalah serial receive shift register. Shift register ini diaktifkan oleh bit receive enable (RE) pada SCI control register 2 (SCCR2). Jika bit ini pada kondisi logika 1 maka data yang diterima melalui pin Rx yang terletak pada buffer akan ditransfer ke shift register.

Bit M pada SCCR1 menentukan apakah shift register akan menerima sepanjang 10 atau 11 bit. Setelah stop bit terdeteksi maka data yang diterima ditransfer ke SCDR dan receive data register full status flag (RDRF) diset pada kondisi logika 1. Ketika data/karakter siap ditransfer ke buffer tapi data atau karakter terdahulu belum terbaca, maka akan terjadi kondisi overrun. Pada kondisi ini data tidak akan ditransfer dan bit overrun status flag (OR) diset pada kondisi logika 1 untuk menunjukkan adanya kesalahan.

Bit control WAKE dari register SCCR1 digunakan untuk memilih apakah menggunakan sinyal MSB (address mark) atau sinyal idle line untuk mengaktifkan receiver. Jika kondisi yang dipilih untuk mengaktifkan receiver terdeteksi maka wake up logic akan menset bit RWU pada register SCCR2 pada kondisi logika 0 yang akan mengaktifkan receiver.



Gambar 2.21*
Blok Diagram Receiver SCI MC68HC11E9

Sistem SCI dikontrol oleh lima buah register, yaitu register BAUD (\$102B), SCDR (\$102F), SCCR1 (\$102C), SCCR2 (\$102D), SCSR (\$102E). Gambar 2.22 menunjukkan isi bit kontrol dari tiap register tersebut.

Register BAUD digunakan untuk memilih baud rate untuk operasi SCI. Register SCCR1 mengandung bit kontrol untuk menentukan panjang format data dan menyeleksi metode yang digunakan untuk mengaktifkan receiver SCI. Register SCCR2 mengandung

BAUD — Baud Rate

\$102B

Bit 7	6	5	4	3	2	1	Bit 0
TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0
RESET:	0	0	0	0	0	U	U

SCCR1 — SCI Control Register 1

\$102C

Bit 7	6	5	4	3	2	1	Bit 0
R8	T8	0	M	WAKE	0	0	0
RESET:	U	U	0	0	0	0	0

SCCR2 — SCI Control Register 2

\$102D

Bit 7	6	5	4	3	2	1	Bit 0
TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET:	0	0	0	0	0	0	0

SCSR — SCI Status Register

\$102E

Bit 7	6	5	4	3	2	1	Bit 0
TDRE	TC	RDRF	IDLE	OR	NF	FE	0
RESET:	1	1	0	0	0	0	0

SCDR — SCI Data Register

\$102F

Bit 7	6	5	4	3	2	1	Bit 0
R7/T7	R6/T6	R5/T5	R4/T4	R3/T3	R2/T2	R1/T1	R0/T0
RESET:	U	U	U	U	U	U	U

Gambar 2. 22⁷
Register-Register Kontrol untuk SCI

bit-bit kontrol utama untuk operasi SCL Empat bit order tinggi dari register ini digunakan untuk mengontrol sinyal interrupt request. Bit TE dan RE digunakan untuk mengaktifkan transmitter dan receiver.

Bit RWU digunakan untuk mengaktifkan receiver. Bit SBK digunakan untuk mengaktifkan sinyal break pada transmitter yaitu bit TDRE dan TC serta lima buah bit status flag untuk receiver yaitu bit RDRF, OR, idle line detect (IDLE), noise flag (NF) dan framing error indication (FE). Register SCDR merupakan dua buah register yang terpisah, yaitu Transmit Data Register (TDR) dan Receive Data Register (RDR). Jika software membaca SCDR berarti RDR yang aktif, jika software menuliskan data pada SCDR berarti TDR yang aktif.

II. 8 PPI (PROGRAMABLE PHERIPHERAL INTERFACE) 8255

PPI 8255 adalah peralatan general purpose programmable I/O (Input/output) yang dapat diprogram secara individual ke dalam dua group dan dioperasikan dalam tiga mode yaitu mode 0, mode 1 dan mode 2.

I. 8. 1 MODE 0 (BASIC I/O)

Konfigurasi ini menyediakan operasi-operasi sederhana untuk input dan output bagi ketiga buah port yang ada. Tidak ada sinyal handshaking yang bisa diberikan ataupun diterima. Data secara sederhana dikirim dan dibaca dari port.

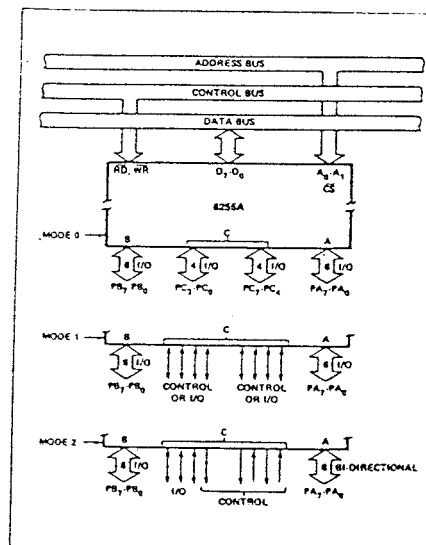
II. 8. 2 MODE 1 (STROBED I/O)

Konfigurasi operasi ini menyediakan fasilitas untuk mentransfer data I/O dari dan ke port tertentu dengan dilengkapi sinyal handshaking. Dalam hal ini port A dan port B

digunakan untuk transfer data, sedangkan port C digunakan untuk sinyal handshaking.

II. 8. 3 MODE 2 (STROBED BIDIRECTIONAL I/O)

Konfigurasi operasi ini menyediakan fasilitas untuk komunikasi data 8 bit 2 arah dengan peralatan luar. Tersedia sinyal-sinyal untuk handshaking dan interrupt dengan fungsi enable dan disable-nya.



Gambar 2. 23^a
Mode Operasi PPI 8255

BAB III

PERENCANAAN PERANGKAT KERAS

III. 1 PENDAHULUAN

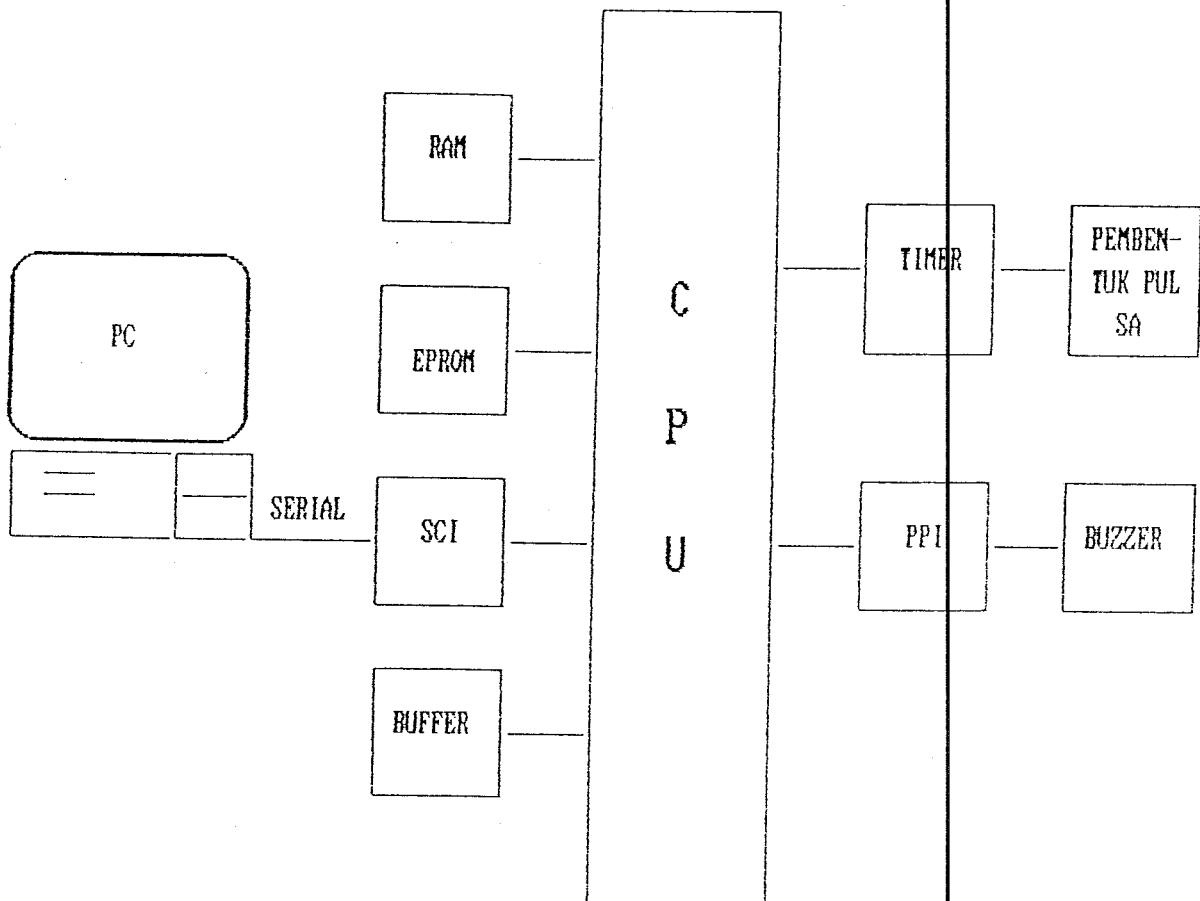
Pada bab ini akan dibahas tentang perencanaan perangkat keras (hardware) dari alat yang dibuat.

Perencanaan perangkat keras yang digunakan meliputi mikrokontroler MC68HC11E9 dalam hal ini berupa EVBU (Evaluation Universal Board) ditambah sistem ekspansi memori yang terdiri dari RAM dan EPROM, sistem data serial, alat pengambil data kecepatan linear (km/jam) dan pengambil data kecepatan rotasi (rpm), dan speed warning yang akan membunyikan buzzer jika kecepatan putarannya melebihi batas maksimal.

III. 2 DIAGRAM BLOK HARDWARE

Bagian ini menjelaskan secara blok diagram seluruh hubungan dari pengambilan data-data oleh mikrokontroler sampai komunikasi serial ke komputer IBM PC.

Bagian blok mikrokontroler terdiri dari sebuah chip tunggal mikrokontroler MC68HC11E9 sebagai CPU dan dilengkapi dengan komponen-komponen pelengkap lainnya seperti rangkaian pembentuk pulsa untuk pengambilan data kecepatan linear dan kecepatan putaran, rangkaian deteksi posisi gigi dan speed warning. Buffer diperlukan



Gambar 3. 1
Blok Diagram Hardware

karena data yang dikirim dari mikrokontroler ke unit lain secara fisik terpisah dan data yang masuk ke mikrokontroler harus selalu valid, agar data yang diolah benar.

Bagian blok memori terdiri dari 1 buah EPROM, 3 buah RAM dan rangkaian decoding. EPROM yang digunakan adalah jenis 2764 (8K) dan untuk RAM digunakan type

6264 (8K) sebanyak 3 buah. Eprom digunakan untuk menyimpan program secara keseluruhan dari mikrokontroler MC68HC11E9 sedangkan RAM digunakan untuk menyimpan data.

Untuk pengambilan data kecepatan dibutuhkan rangkaian pembangkit pulsa yang diperoleh dari piringan berlubang. Pulsa-pulsa ini diberikan pada port A (ICx) dari mikrokontroler. Untuk speed warning dibutuhkan 4 buah port I/O, untuk itu maka ditambah PPI 8255.

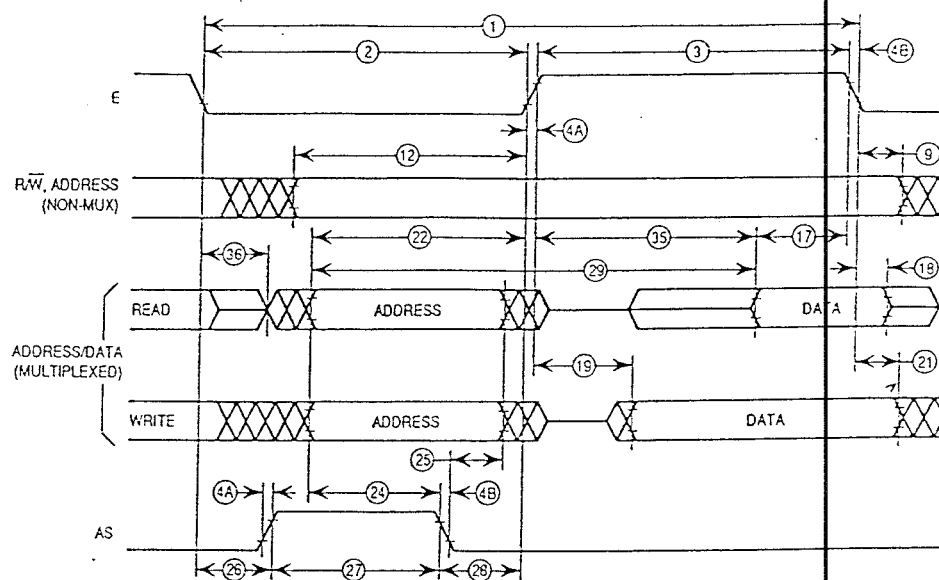
III. 3 UNIT PERANGKAT KERAS

Perangkat keras yang digunakan adalah MC68HC11E9 sebagai pengendali sistem dengan mode EXPANDED MULTIPLEXED (menggunakan RAM dan EPROM luar). Sebagai Input/Output digunakan Programmable Peripheral Interface (PPI) 8255.

III.3.1 CENTRAL PROCESSING UNIT

Central Processing Unit (CPU) dibentuk oleh mikrokontroler MC68HC11 yang dioperasikan pada expanded mode. Pada mode ini, MC68HC11 memberikan byte high dari bus alamat pada port B. Sedangkan byte low dari bus alamat dimultiplex bersama bus data pada port C, dimana byte low pada bus alamat akan diberikan pada setengah siklus pertama bersamaan dengan sinyal Address Strobe (AS) yang aktif high dan bus data pada setengah siklus berikutnya.

Diagram waktu dari mode expanded diberikan pada gambar 3.2.



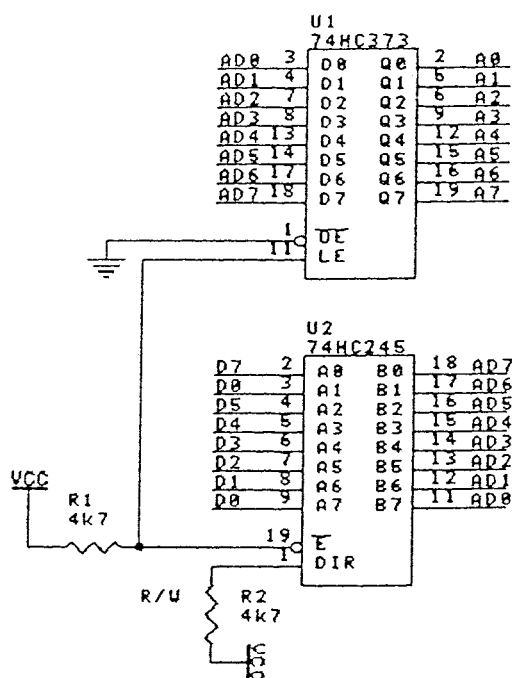
NOTE: Measurement points shown are 20% and 70% of V_{DD} .

Gambar 3.2
Diagram Waktu Mode Expanded

Untuk memisahkan byte low pada bus alamat dan bus data maka digunakan rangkaian demultiplexer seperti terlihat pada gambar 3.3.

Pada rangkaian demultiplexer digunakan 74HC373 untuk me-latch byte low bus alamat. Input G dari 74LS373 diperoleh dari Address Strobe (AS) dan Output Control-nya (OC) dihubungkan ke ground sehingga output 74LS373 selalu terhubung ke bus alamat.

Bus data terhubung ke port C MC68HC11 melalui 74HC245. Enable (G) dari 74HC245 diperoleh dari inverter sinyal E. Sinyal E akan high pada saat MC68HC11 melakukan akses ke data bus, sehingga 74HC245 akan di-enable pada saat tersebut. Arah data pada 74HC245 oleh input Direction (DIR), yaitu dari input B ke output A jika input DIR pada kondisi low dan sebaliknya dari input A ke output B jika input DIR pada kondisi high.



Gambar 3.3
Rangkaian Demultiplexer

Karena input/output A dari 74HC245 terhubung ke port C dan input/output B terhubung ke bus data, maka input DIR-nya diperoleh dari sinyal Read (RD), 74HC245 akan dienable pada arah input B ke output A pada saat operasi Read dari bus data ke MC68HC11 dan pada arah sebaliknya pada saat operasi Write.

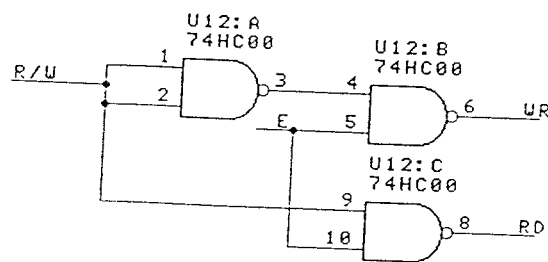
Sinyal Read (RD) dan Write (WR) diperoleh dengan men-dekode-kan sinyal E dan R/W dari MC68HC11. Kondisi sinyal E dan R/W pada saat operasi Read dan Write dapat dilihat tabel 3.1.

Tabel 3.1
Sinyal E dan R/W

E	R/W	
0	x	Proses Internal
1	1	Proses Baca (Read)
0	x	Proses Internal
1	0	Proses Tulis (Write)

x = don't care

Berdasarkan tabel 3.1 dibuat rangkaian decoder untuk sinyal kontrol seperti terlihat pada gambar 3.4.



Gambar 3.4
Decoder E dan R/W

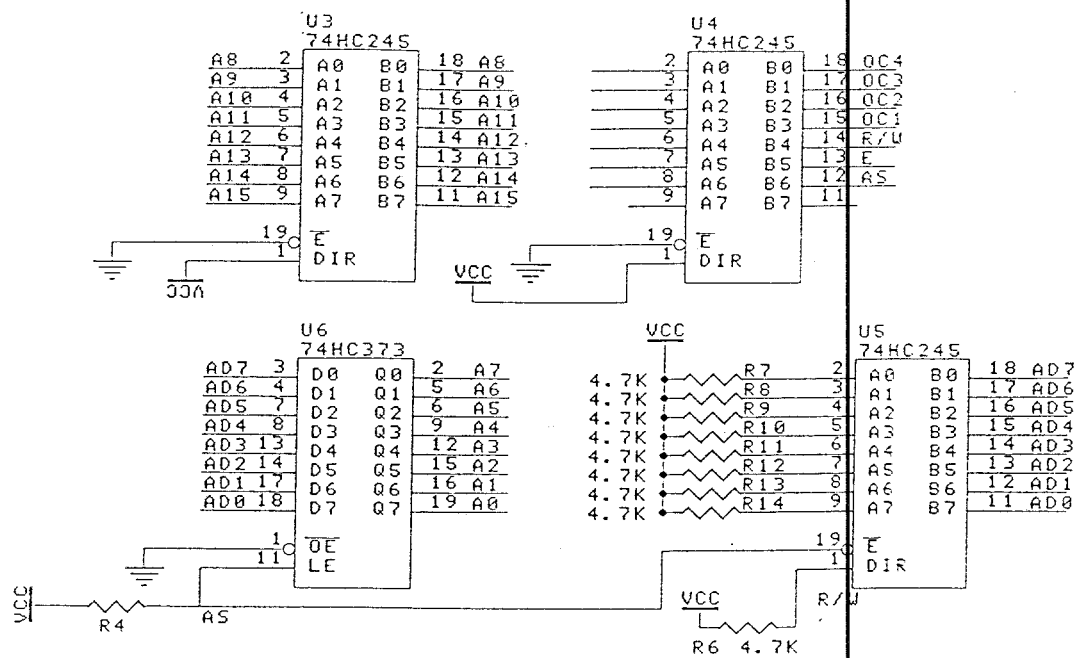
Output RD dan WR adalah aktif low. Pada saat E berkondisi low dimana terjadi proses internal, maka kedua output RD dan WR tidak ada yang aktif. Setelah E berkondisi high, yaitu pada saat proses eksternal, salah satu output akan aktif bergantung

padakondisi R/W. Jika R/W berkondisi high, maka output RD akan low. Sebaliknya, jika R/W berkondisi low pada saat E high, maka output yang low.

Rangkaian lengkap demultiplexer dan dedecoder sinyal RD dan WR diberikan pada lampiran.

III. 3. 2 RANGKAIAN BUFFER MIKROKONTROLOR MC68HC11

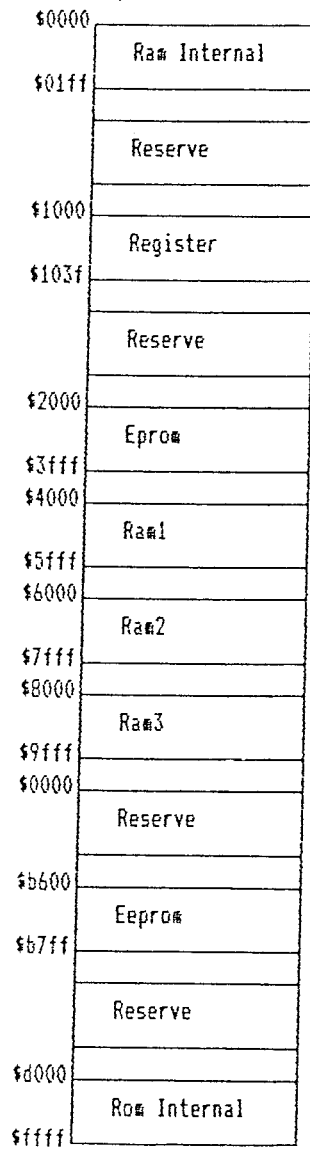
Dari unit mikrokontroler yang merupakan CPU, maka untuk port data dan alamat yang keluar atau masuk mikrokontroler perlu diberi buffer yang dapat dilihat pada gambar 3.5.



Gambar 3.5
Rangkaian Buffer Data dan Alamat

III. 3. 3 MEMORY MAPPING

Perencanaan memory mapping diperlihatkan pada gambar 3.6.



Gambar 3.6
Perencanaan Memori mapping

III. 3. 4 RANGKAIAN MEMORI EKSTERNAL

Seperti terlihat pada gambar 2.12 pada bab II daerah yang diarsir menunjukkan lokasi memori internal, sedang yang bertanda EXT adalah lokasi yang bisa diakses oleh memori eksternal.

Untuk itu dipilih alamat memori eksternal sebagai berikut:

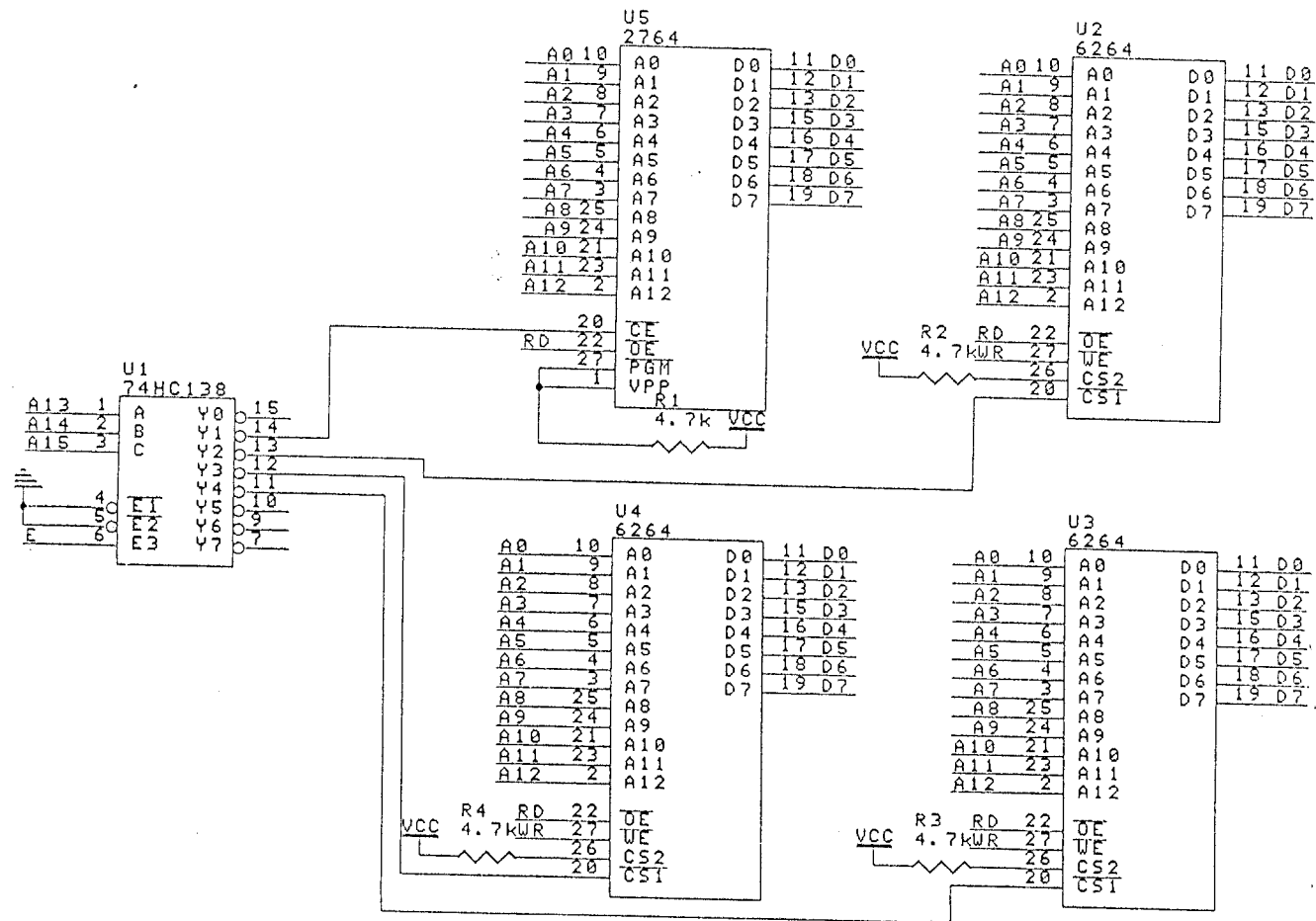
- ♦ EPROM 2764 (8 Kbyte) dengan alamat \$2000-\$3FFF.
- ♦ 3 buah RAM 6264 (8 Kbyte) dengan alamat \$4000-\$5FFF; \$6000-\$7FFF ;
\$8000-\$9FFF.

Untuk pengalamatan tersebut diperlukan rangkaian decoding dengan tabel kebenaran seperti pada tabel 3.2. Rangkaian memori eksternal ini dapat dilihat pada gambar 3.7 dan untuk lebih lengkapnya dapat dilihat pada lampiran

Tabel 3.2
Decoding RAM dan EPROM

MEMORI	ALAMAT	A15	A14	A13	A12
2764	\$2000-\$3fff	0	0	1	x
2664(1)	\$4000-\$5fff	0	1	0	x
6264(2)	\$6000-\$7fff	0	1	1	x
6264(3)	\$8000-\$9fff	1	0	0	x

x = don't care

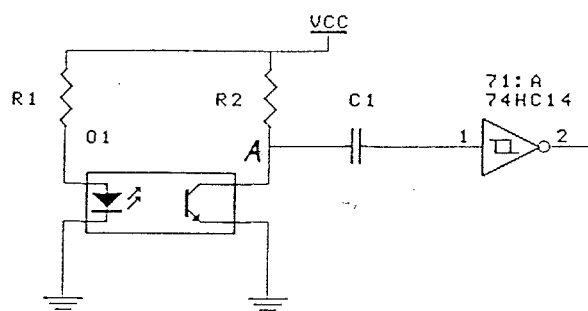


Gambar 3.7
Rangkaian Memori Eksternal

III. 3. 5 DETEKTOR KECEPATAN LINEAR (KM/H)

Detektor kecepatan disini dibuat secara sederhana dengan menggunakan lempeng aluminium yang dilubangi pinggirnya yang diputar oleh motor dc.

Sebagai sensor digunakan optocoupler yang diaktif tidakkan oleh lubang bergerigi aluminium sehingga dapat menghasilkan pulsa. Rangkaian pembentuk pulsa ini dapat dilihat pada gambar 3.8.



Gambar 3.8
Rangkaian Pembentuk Pulsa

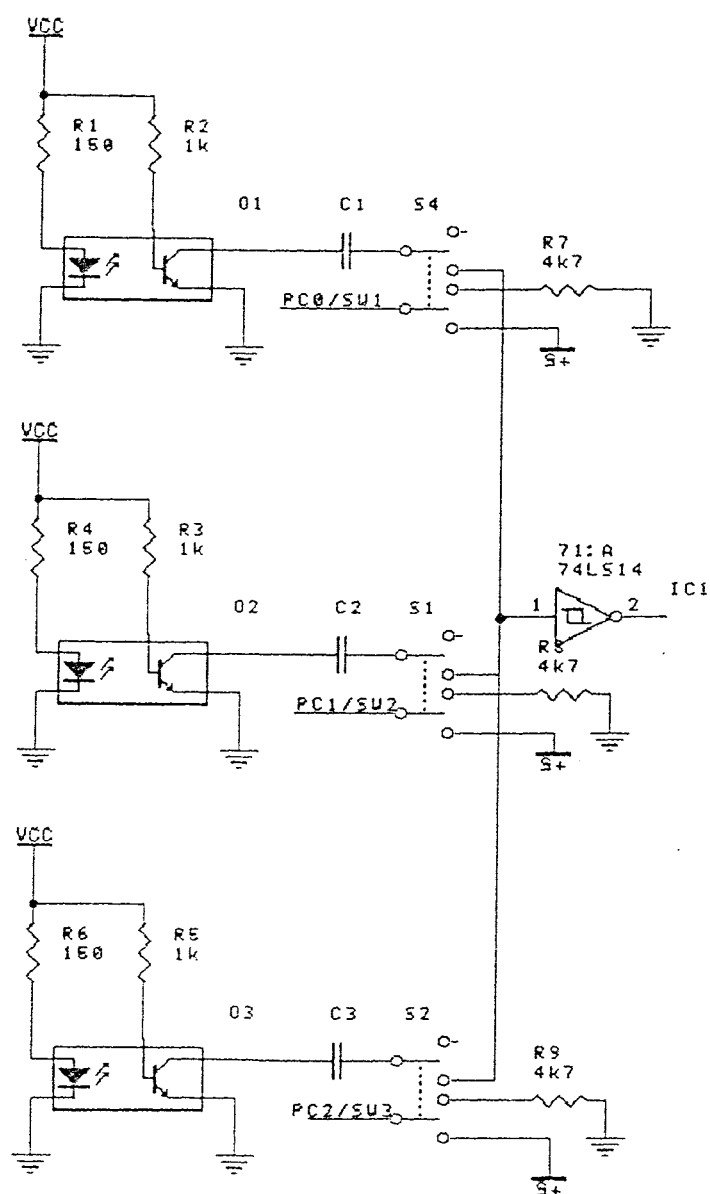
Dari rangkaian ini dapat dilihat jika lubang pada aluminium mengenai optocoupler maka transistor akan saturasi, level tegangan pada titik a akan menjadi low, dan dengan menggunakan schmitt trigger 74HC14 maka level tegangan ini akan diinverter sehingga menjadi high.

III. 3. 6 DETEKTOR KECEPATAN PUTARAN (RPM)

Untuk detektor kecepatan putaran juga menggunakan rangkaian yang sama dengan detektor kecepatan linear, hanya saja untuk kecepatan putaran ini digunakan tiga buah sensor optocoupler dan tiga buah lempeng yang tidak sama besarnya yang berfungsi untuk

gigi persneling. Lempeng terkecil untuk gigi rendah, dan lempeng terbesar untuk gigi tertinggi.

Rangkaian detektor untuk kecepatan rpm dapat diperlihatkan pada gambar 3.9.



Gambar 3.9
Rangkaian Detektor Untuk Kecepatan Rpm

III.3. 7 RANGKAIAN WARNING RPM dan DETEKSI POSISI GIGI

Jika kecepatan putaran rpm melebihi batas yang ditentukan maka gigi harus dinaikkan, jika tidak maka buzzer akan berbunyi. Untuk deteksi posisi gigi ini digunakan switch dan untuk warning rpm dibutuhkan 4 buah port input/output. Untuk ini maka digunakan PPI 8255 dengan alamat \$200-\$203. Port A dengan alamat \$200, port B \$201, port C \$202, dan port Control dengan alamat \$203. Berikut tabel 3.3 merupakan pengalamatan yang digunakan untuk decoding PPI

Tabel 3. 3
Decoding Pengalamatan PPI 8255

Alamat	a15	a14	a13	a12	a11	a10	a9	a8	a7	a6	a5	a4	a3	a2	a1	a0
\$200-\$203	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x

Dalam rangkaian ini digunakan decoding penuh dan untuk lebih jelasnya rangkaian lengkap warning rpm dan deteksi posisi gigi ini dapat dilihat pada gambar 3.10.

BAB IV

PERENCANAAN PERANGKAT LUNAK

IV.1 PENDAHULUAN

Perangkat lunak merupakan bagian dari perancangan prototipe sistem. Secara keseluruhan perangkat lunak yang mendukung bekerjanya sistem terdiri dari dua bagian yaitu perangkat lunak untuk mikrokontroler MC68HC11E9 dan perangkat lunak untuk komputer IBM PC. Perangkat lunak untuk mikrokontroler menggunakan Cross Assembler MC68HC11E9 dan untuk komputer digunakan bahasa PASCAL.

Komputer IBM PC digunakan untuk mengolah dan menampilkan data-data yang tersimpan di RAM dalam bentuk grafik. Grafik ini meliputi grafik kecepatan linear terhadap waktu, grafik kecepatan rotasi terhadap waktu. Data dari RAM ini dikirim secara serial ke IBM PC, kemudian diolah untuk ditampilkan dalam bentuk grafik.

IV.2 PERANGKAT LUNAK PADA MIKROKONTROLER

MC68HC11E9

Dalam perencanaan perangkat lunak untuk MC68HC11E9 dibagi menjadi beberapa langkah:

1. Inisialisasi sistem dan komunikasi serial.
2. Proses pengambilan data per detik dan pemberian peringatan pada pengemudi jika kecepatannya melebihi batas yang diijinkan.

Proses pengambilan data ini dilakukan secara serentak. Jika kecepatan putaran melebihi batas yang diijinkan maka gigi harus dinaikkan dan buzzer akan berbunyi sebagai peringatan. Demikian juga untuk kecepatan linear jika melebihi batas maka buzzer juga akan berbunyi. Proses pengambilan data ini berlangsung secara terus-menerus sampai akhir perjalanan (dalam hal ini sampai waktu tertentu sesuai dengan kapasitas RAM yang digunakan).

Setelah proses pengambilan data selesai berikutnya akan dilakukan pengecekan apakah sudah dihubungkan dengan komputer (komputer siap menerima data). Jika sudah siap maka MCU MC68HC11E9 akan mengirimkan data ke IBM PC.

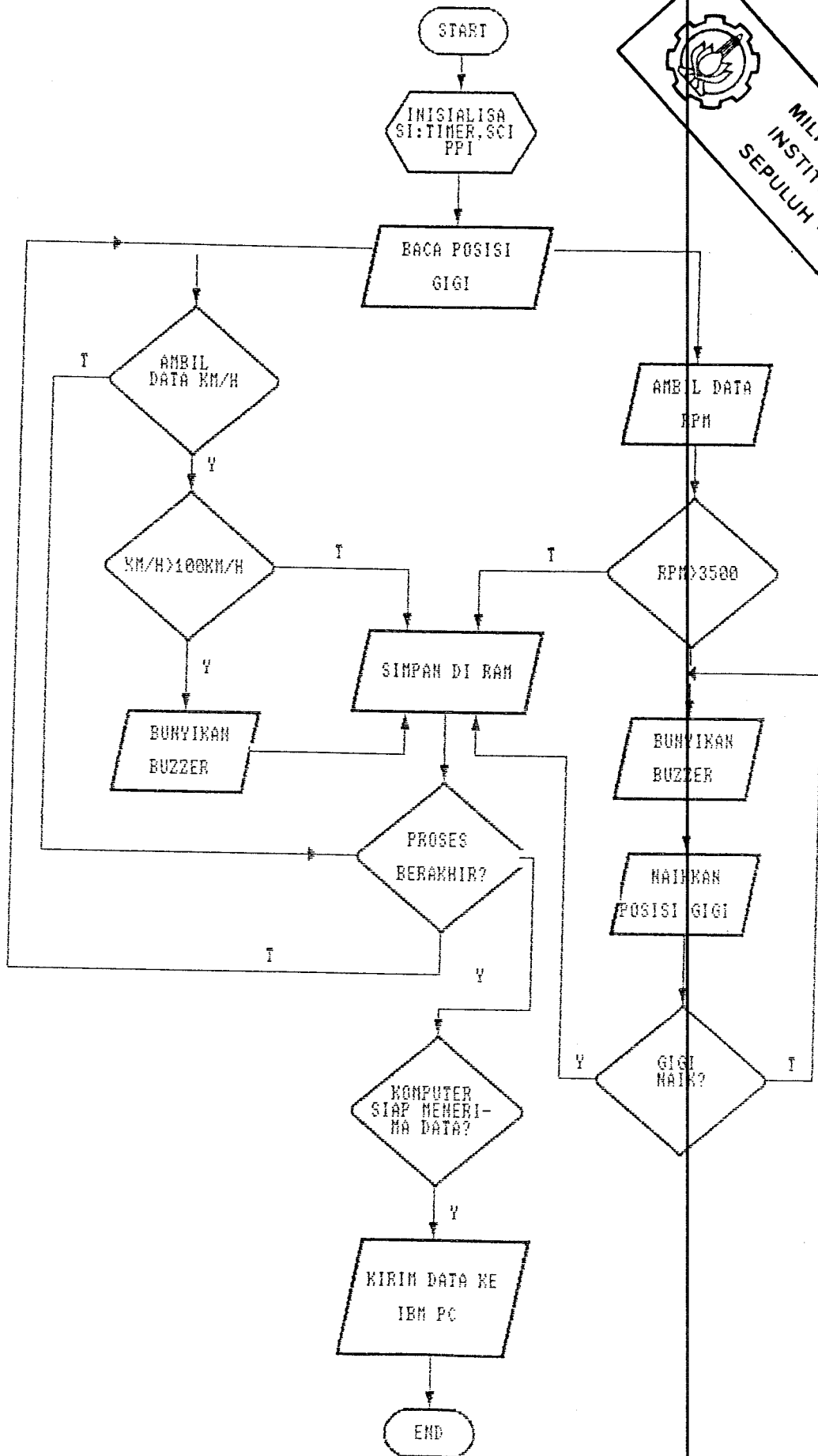
Diagram alir program secara umum dapat dilihat pada gambar 4.1

IV. 2. 1 ROUTINE-ROUTINE PERANGKAT LUNAK MIKROKONTROLER

MC68HC11E9

IV.2. 1. 1 ROUTINE INISIALISASI

1. Menentukan alamat awal untuk stack dalam hal ini posisi teratas dari stack diletakkan pada alamat \$9FFF yang merupakan alamat terakhir dari RAM eksternal.
2. Menginisialisasi Register Timer Control 2 (TCTL2) seperti terdapat dalam gambar 4. 2.
3. Menginisialisasi register BAUD seperti pada gambar 4.3



Gambar 4.1
Diagram Alir Program Mikrokontroler

TCTL2 - Timer Control 2

Bit 7	6	5	4	3	2	1	Bit 0	\$1021
EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	
Reset :	0	0	0	0	0	0	0	0

Gambar 4.2
Inisialisasi TCTL2

BAUD - Baud Rate

Bit 7	6	5	4	3	2	1	Bit 0	\$102B
TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0	
Reset :	0	0	0	0	0	U	U	U

Gambar 4.3
Inisialisasi Baud

IV. 2. 1. 2 ROUTINE TIMER

Untuk menghitung kecepatan motor digunakan timer pada MC68HC11E9, dalam hal ini Input Capture 1 (IC1) dan Input Capture 2 (IC2). Untuk menghitung kecepatan linear digunakan IC2 dan IC1 digunakan untuk menghitung kecepatan putaran. Dari timer ini dapat dihitung periode kecepatan motor. Berikut diberikan contoh routine timer untuk IC1:

```
ldaa  #%00010000
staa  TCTL2,x
ldaa  #$04
staa  TFLG1,x      ; clear any old ocl flag
brclr TFLG1,x,$04 * ; loop here until edge
ldd   TIC1,x       ; read time of first edge
std   frst1        ; save first capture value
ldaa  #$04
staa  TFLG1,x      ; clear ic1f before next edge
```

```

brclr  TFLG1,X,#$04 *   ; loop here until edge
ldd    TIC1,X           ; read time of second edge
subd   first1           ; 2nd- 1st --> d
std    percl            ; save result (period in cys)

```

Dari routine ini dapat dijelaskan bahwa pertama kali kita menentukan IC mana yang akan dipakai. Dalam hal ini kita menggunakan IC1 dan capture hanya terjadi pada rising edge. Untuk itu kita perlu mengeset register TFLG1 dan TCTL2 sebagai inisialisasi.

Selanjutnya akan dilakukan pembacaan terhadap sisi yang pertama terdeteksi, dan register TFLG1 akan dideclear sampai sisi kedua terdeteksi. Untuk mendapatkan periode maka dilakukan pengurangan sisi kedua terhadap sisi pertama.

IV. 2 1.3 ROUTINE SERIAL

Data yang telah tersimpan di RAM selanjutnya dikirimkan secara serial ke IBM

PC. Adapun routine inisialisasi serial adalah sebagai berikut:

```

onsci  ldaa  #$30
        staa  BAUD
        ldaa  #$00
        staa  $SCCR1
        ldaa  #$0C
        staa  SCCR2
        rts

```

Pengiriman dilakukan dengan membaca register SCSR dan dilakukan test dengan bit \$80 dan selanjutnya hasilnya disimpan dalam register SCDR sebagai data register. Adapun routine yang digunakan adalah sebagai berikut :

```

kirim  pshx
        psha
        pshb
lp      ldab  SCSR
        andb  #$80

```

```

beq    lp
ldaa   out
staa   SCDR
pulb
pula
pulx
rts

```

IV. 2. 1. 4 ROUTINE INPUT/OUTPUT

PPI 8255 digunakan untuk speed warning karena rangkaian ini memerlukan 4 buah I/O port. Alamat yang dipakai adalah \$200-\$203. Routine Assembler MC68HC11 untuk inisialisasi PPI 8255 adalah sebagai berikut:

```

Init_PPI  ldaa  #$89
          staa  $203
          rts

```

IV. 3 PERANGKAT LUNAK PADA KOMPUTER IBM PC

Program pendukung pada IBM PC secara garis besar mengerjakan beberapa hal sebagai berikut :

- ♦ Penanganan komunikasi data serial
- ♦ Pengolahan data dan tampilan.

IV. 3. 1 KOMUNIKASI DATA SERIAL

Program ini merupakan inisialisasi port serial dengan menggunakan interrupt 14h fungsi 0, dengan format sebagai berikut :

- ♦ 1 start bit, 8 bit data, no parity dan 1 stop bit dengan laju transmisi 9600 baud.

Berikut diberikan inisialisasi untuk komunikasi serial:

```
Procedure InitCom;  
  R : registers;  
  R.Dx := 0;  
  R.A1 := $E3;  
  R.AH := 0;  
  BIntr ($14,R);  
End;
```

IV. 3. 2 PENGOLAHAN DATA DAN TAMPILAN

Proses pengolahan data ini meliputi :

- ♦ Penyimpanan data dalam bentuk file.
- ♦ Menampilkan data dalam bentuk grafik.

Tampilan meliputi menu yang terdapat dilayar monitor berupa:

- ♦ Listing data kecepatan baik kecepatan linear (km/h) maupun kecepatan putaran (rpm).
- ♦ Grafik kecepatan linear terhadap waktu dan grafik kecepatan putaran terhadap waktu dalam satu sumbu grafik. Print untuk listing data dan grafik.

BAB V

PENGUJIAN DAN PENGUKURAN

V. 1 PENGAMBILAN DATA

Pengambilan data dilakukan dalam selang waktu tertentu secara periodik. Hasil pengambilan data lengkap dapat dilihat pada tabel 5.1.

Tabel 5.1

Pengambilan data	F(km/h)Hz	F(rpm)Hz
I	39	34
	39	35
	40	35
	39	36
	39	36
	39	36
	39	36
Rata-rata	39,14	35,28
II	49	44
	49	44
	49	45
	49	44
	49	44
	49	44
	49	44
Rata-rata	49	44,14
III	65	57
	64	57
	64	57

Pengambilan data	F(km/h)Hz	F(rpm)Hz
III	63	58
	63	58
	63	57
	63	57
Rata-rata	63,71	57,28
IV	83	71
	85	110
	85	110
	85	74
	85	74
	85	74
	85	73
Rata-rata	84,71	83,71

Setelah dilakukan pengambilan data maka berikutnya data-data ini diolah untuk ditampilkan dalam bentuk grafik.

Grafik ini berupa grafik kecepatan linear dan kecepatan putaran terhadap waktu dalam satu sumbu grafik.

V. 2 ANALISIS DATA

Data kecepatan linear (km/h) diperoleh dari pulsa yang dihasilkan oleh piringan yang berlubang yaitu dengan melakukan pengukuran terhadap frekwensinya. Dengan mengatur tegangan motor maka akan diperoleh data kecepatan yang berbeda-beda. Kecepatan linear didapat dengan mengalikan frekwensi dengan suatu konstanta tertentu. Konstanta ini diperoleh dari perhitungan sebagai berikut:

$$V = 2 \pi r F \text{ m/det}$$

dimana $r = \text{jari-jari piringan} = 5.5 \text{ cm} = 0,055\text{m}$.

$$V = 2 \pi F \times (0,055)\text{m/det}$$

$$= 0,3454 F \text{ m/det}$$

jika diubah kedalam satuan km/jam maka:

$$V = 0,3454 \times \frac{3600}{1000} \times F = 1,244 \times F \text{ km/jam}$$

$$\text{Kecepatan linear} = V = k \times F$$

Dimana $k = \text{konstanta} = 1,244$.

$F = \text{frekwensi piringan}$.

Karena dalam hal ini kecepatan linear dan kecepatan putaran disesuaikan dengan kondisi bis (mercedes benz) semua data-data frekwensi diatas dikonversi dan disesuaikan dengan data yang ada. Misal untuk frekwensi $\text{km/h} = 85\text{hz}$ maka kecepatan akan sama dengan :

$$V (\text{km/h}) = 1,244 \times 85 = 105,74 \text{ km/h}$$

Sedangkan data kecepatan maksimal pada kendaraan mercedes benz adalah 125 km/h, maka konstanta pengukuran perlu diperbesar yaitu sebagai berikut :

$$K' = 1,244 \times 125/105,74 = 1,4756$$

sehingga harga $V (\text{kmh})$ adalah :

$$V(\text{kmh}) = 1,4756 \times F(\text{hasil pengukuran})$$

Hasil perhitungan seluruh data dapat dilihat pada tabel 5.2

Demikian juga untuk data rpm

$$\text{Rpm} = 2 \pi \times F \times 60$$

$$= 120 \pi \times F$$

Untuk $F \text{ (rpm)} = 34 \text{ Hz}$ didapat :

$$\text{Rpm} = 2\pi \times 34 \times 60 = 2040$$

Untuk jenis kendaraan marcedes benz Rpm minimum yang baik adalah 1500, untuk ini data perlu disesuaikan sehingga didapat :

$$\text{Rpm} = \frac{F(\text{rpm})}{34 \times 1500}$$

Hasil perhitungan lengkap dapat dilihat pada tabel 5.2

Tabel 5.2

Pengambilan data	F(km/h)Hz	F(rpm)Hz	V(km/h)	Rpm
I	39	34	57,22	1500,0
	39	35	57,22	1544,1
	40	35	59,00	1544,1
	39	36	57,22	1588,2
	39	36	57,22	1588,2
	39	36	57,22	1588,2
	39	36	57,22	1588,2
Rata-rata	39,14	35,28	57,40	1556,4
II	49	44	72,3	1941,2
	49	44	72,3	1941,2
	49	45	72,3	1985,2
	49	44	72,3	1941,2
	49	44	72,3	1941,2
	49	44	72,3	1941,2
	49	44	72,3	1941,2
Rata-rata	49	44,14	72,3	1947,3
III	65	57	95,91	2514,6
	64	57	94,44	2514,6
	64	57	94,44	2514,6
	63	58	92,96	2558,7
	63	58	92,96	2558,7
	63	58	92,96	2558,7
	64	57	94,4	2514,6
Rata-rata	63,71	57,28	94,01	2526,9

Pengambilan data	F(km/h) Hz	F(rpm)Hz	V(km/h)	Rpm
IV	83	71	122,47	3132,3
	85	110	125,43	4852,8
	85	110	125,43	4852,8
	85	74	125,43	3264,6
	85	74	125,43	3264,6
	85	74	125,43	3264,6
	85	74	125,43	3264,6
Rata-rata	84,71	83,71	125	3692,9

Keterangan:

$$V \text{ (km/h)} = F \text{ (km/h) pengukuran} \times 1,4756$$

$$\text{RPM} = \frac{F(\text{rpm})\text{pengukuran} \times 1500}{34}$$

Dari tabel diatas dapat dibuat grafik dengan pendekatan secara regresi linear, dimana garis lurus hasil regresi data hasil pengukuran diatas dapat dicari dengan persamaan berikut:

$$Y = mx + b$$

$$\text{dimana : } m = \frac{N \sum xy - \sum x \sum y}{\sum x^2 - (\sum x)^2}$$

$$b = \frac{N \sum x^2 - \sum xy \sum x}{\sum x^2 - (\sum x)^2}$$

N = jumlah data

Dari perhitungan diatas maka untuk persamaan garis km h didapat $m = 1,48415$ dan $b = -0,45$ sehingga didapat persamaan garis:

$$V(\text{kmh}) = 1,48415 f(\text{km/h}) - 0,45$$

Untuk persamaan rpm didapat harga $m = 44,115$ dan $b = 0,04$ sehingga persamaan garis rpm adalah:

$$\text{Rpm} = 44,115 f(\text{rpm}) + 0,04$$

V.3 KETERANGAN GRAFIK

Untuk kendaraan jenis mercedes benz terdapat beberapa daerah kerja untuk rpm. Adapun daerah kerja itu adalah sebagai berikut:

- ♦ 1500 rpm s/d 1800 rpm, daerah putaran motor ekonomis yang terbaik.
- ♦ 1800 rpm s/d 2500 rpm, putaran motor ekonomis.
- ♦ 2500 rpm s/d 3000 rpm merupakan putaran motor maksimal.
- ♦ 3000 rpm s/d 3500 rpm merupakan putaran motor yang berlebihan.

Dari grafik ini dapat dilihat data-data kecepatan baik kecepatan linear maupun kecepatan putaran. Dari grafik ini juga dapat dilihat untuk rpm yang baik terdapat pada pengambilan data I, data II merupakan putaran motor ekonomis, data III merupakan putaran motor maksimal dan data IV adalah putaran motor yang harus dihindari dan akan menyebabkan alarm berbunyi sebagai peringatan pada pengemudi.

BAB VI

KESIMPULAN DAN SARAN

Bab ini adalah bab terakhir yang berisi kesimpulan, saran dan kemungkinan pengembangan sistem.

VI.1 KESIMPULAN

1. Untuk mendeteksi kecepatan kendaraan umum selama diperjalana maka parameter yang perlu diambil adalah:
 - ♦ kecepatan linear.
 - ♦ kecepatan putaran (rpm).
2. Data yang telah tersimpan di RAM ini dapat dikirim ke komputer secara serial untuk dibuat grafik dan listing datanya .
3. Dari grafik ini pemilik kendaraan dapat mengetahui bagaimana pengemudi menjalankan kendaraanya.

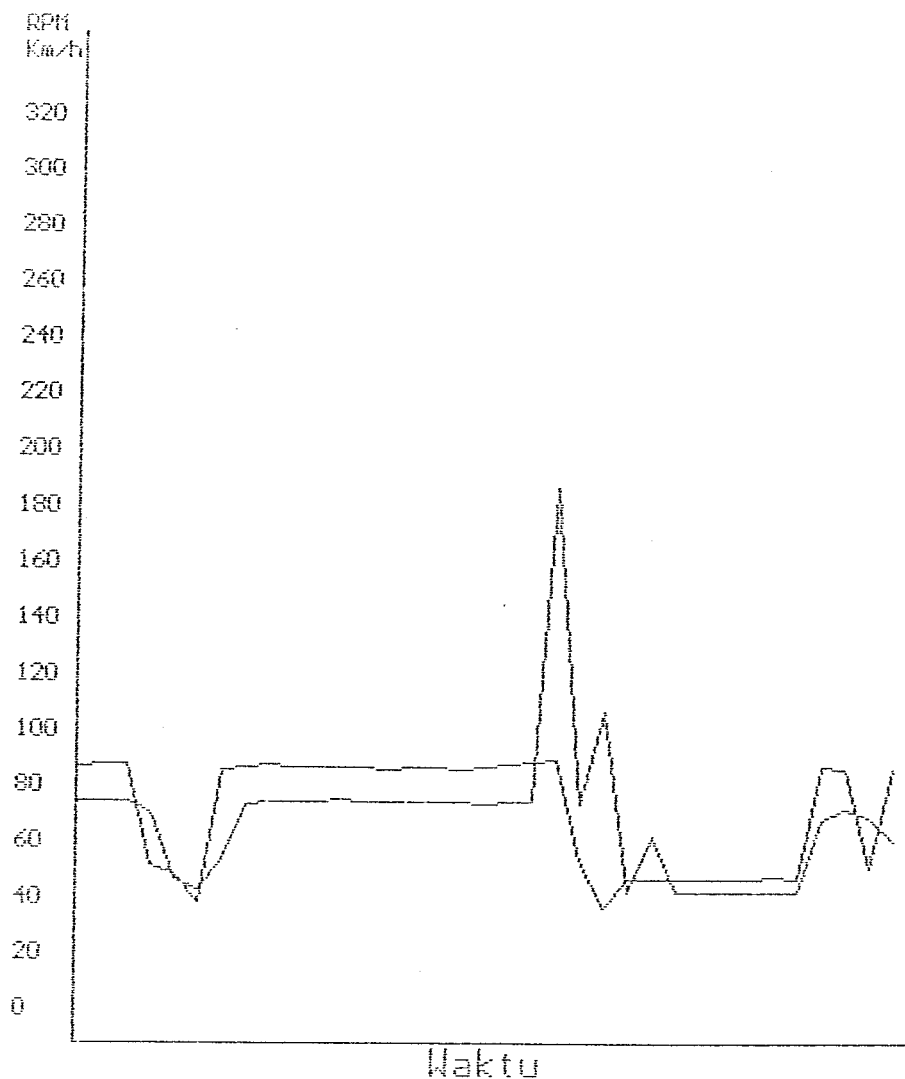
VI. 2 SARAN DAN KEMUNGKINAN PENGEMBANGAN

Prototipe sistem yang telah dibuat pada tugas akhir ini bisa lebih diperbaiki lagi performansinya dan dikembangkan untuk fungsi yang lebih luas. Berikut ini beberapa saran untuk memperbaiki dan mengembangkan sistem ini :

- ♦ Data yang diambil bukan hanya data kecepatan tetapi juga dilengkapi dengan pengukuran suhu , tekanan udara, tekanan pelumasan motor dan pengukur isi tangki bahan bakar .
- ♦ Memori yang digunakan hendaknya diperbesar agar dapat digunakan untuk jangka waktu yang lebih lama dan frekwensi pengambilan datanya lebih cepat.
- ♦ Sebaiknya peralatan ini ditempatkan dalam suatu kotak yang dikemas secara aman (tahan benturan, tahan api) sehingga menyulitkan pengemudi untuk merusak peralatan ini.

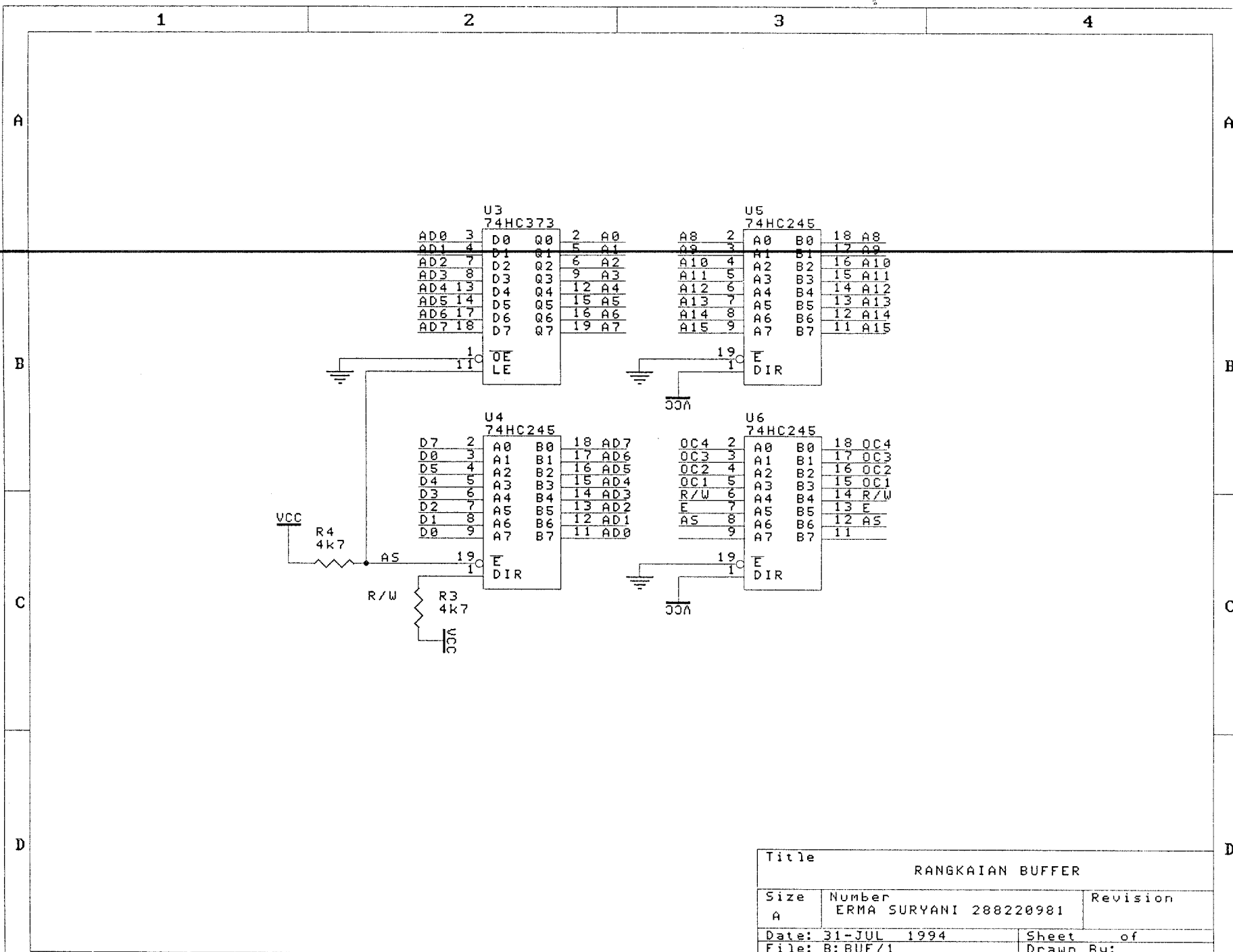
DAFTAR PUSTAKA

1. Boentarto, Drs., *Kelistrikan Mobil*, Andi Offset, Yog-karta, 1993.
2. Peatman, John B., *Design With Microcontroller*, McGraw Hill, Inc., 1988.
3. Steeman, J.P.M., *DATA SHEET BOOK 2*, PT Elex Media Kom-putindo Kelompok Gramedia, Jakarta, 1989.
4. Sutrisno, *Fisika Dasar Mekanika*, ITB, 1982.
5. Lembaga Bengkel HMM-ITS, *OTOMOTIF*, ITS, 1994.
6. Motorola Inc., *MC68HC11E9 HCMOS Single-Chip Microcontroller*, Motorola Inc., 1988.
7. Motorola Inc., *MC68HC11E9 Refrence Manual Revision 2*, Motorola Inc., 1988.
8. Motorola Inc., *MC68HC11EVBU Universal Evaluation Board User's Manual*, Motorola Inc., 1988.
9. Texas Instrument Inc., *The TTL Data Book for Design Engineer*, second edition, Texas Instrumen Inc., 1990

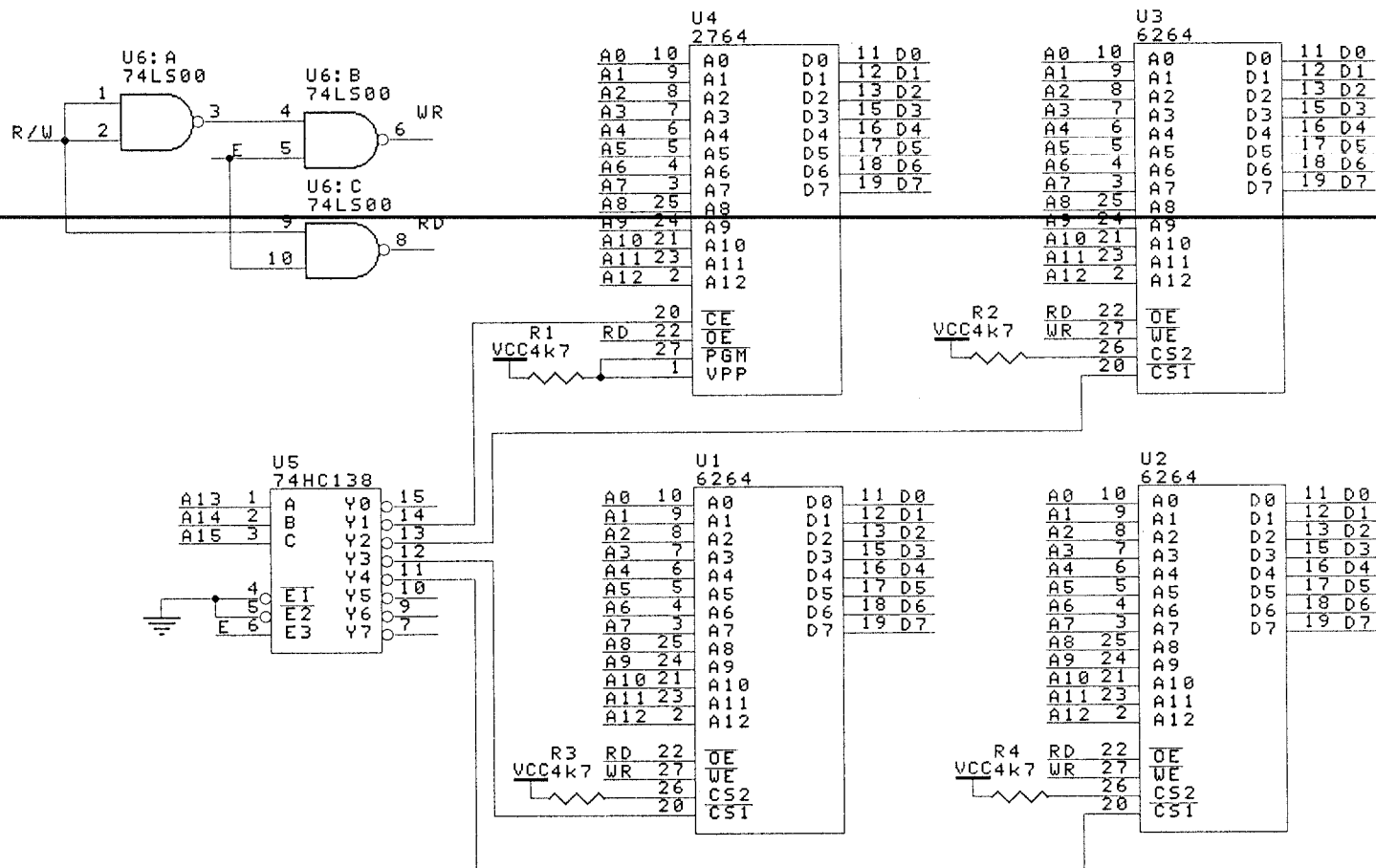


TABEL KECEPATAN DAN PUTARAN
tanggal: Selasa, 2/8/1994

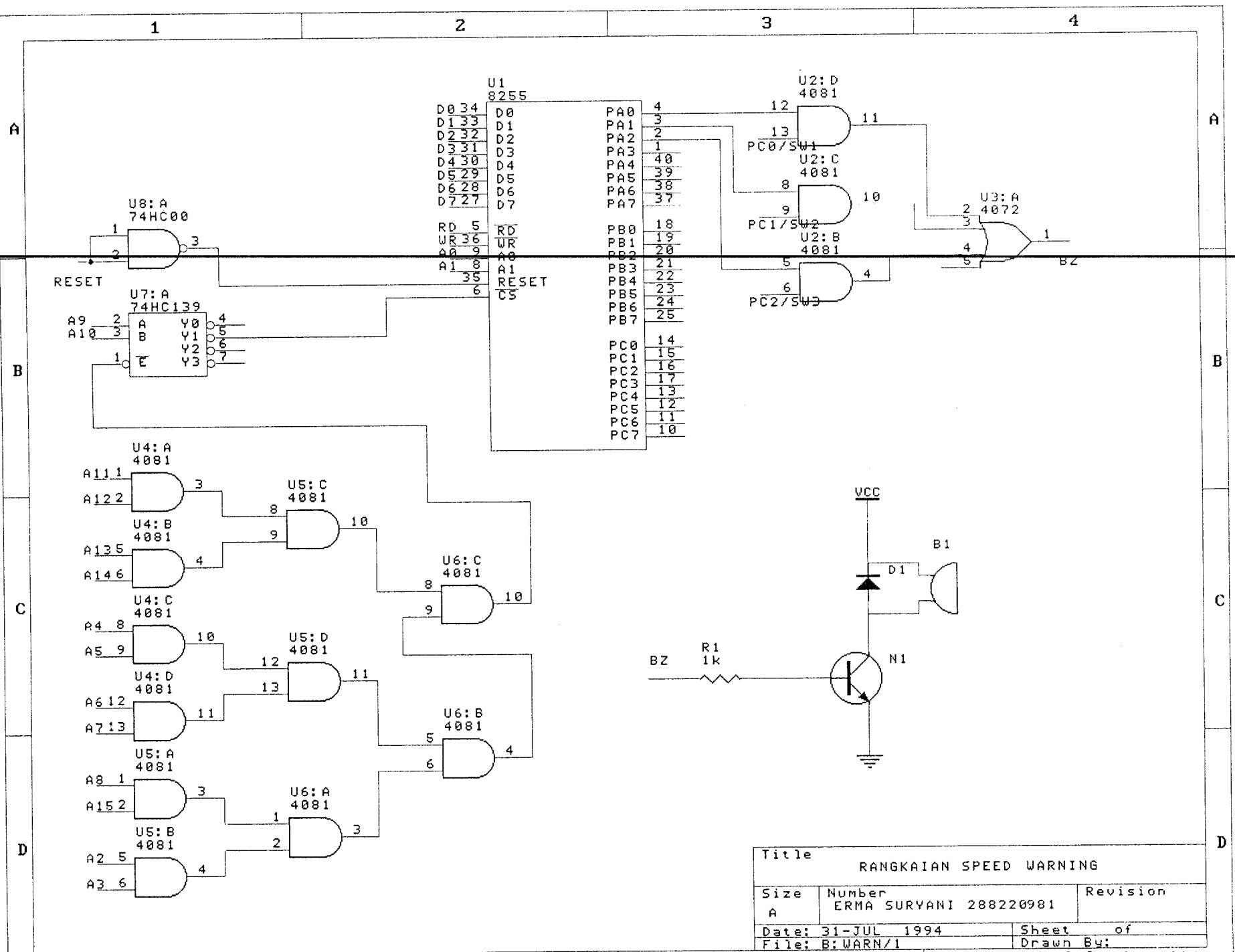
Jam	Putaran (Rpm)	Kecepatan (Km/h)
11:05:40	824.44	48.50
11:05:41	1151.11	44.00
11:05:42	1166.67	44.50
11:05:43	1166.67	44.00
11:05:44	1166.67	44.00
11:05:45	1166.67	44.00
11:05:46	1166.67	44.00
11:05:47	1182.22	44.00
11:05:48	1166.67	44.00
11:05:49	1166.67	44.00
11:05:50	1166.67	44.00
11:05:51	1166.67	43.50
11:05:52	1166.67	44.00
11:05:53	1166.67	44.00
11:05:54	1166.67	44.00
11:05:55	1166.67	44.00
11:05:56	1166.67	43.50
11:05:57	1166.67	44.00
11:05:58	1151.11	44.00
11:05:59	1166.67	44.50
11:06:00	1166.67	44.00
11:06:01	1166.67	45.00
11:06:02	3002.22	45.00
11:06:03	1151.11	45.00
11:06:04	1151.11	24.50
11:06:05	1695.56	17.50
11:06:06	637.78	21.00
11:06:07	637.78	21.00
11:06:08	964.44	21.00
11:06:09	637.78	21.00
11:06:10	637.78	21.00
11:06:11	637.78	21.00
11:06:12	637.78	21.00
11:06:13	637.78	21.00
11:06:14	637.78	21.00
11:06:15	637.78	21.00
11:06:16	637.78	21.00
11:06:17	637.78	21.50
11:06:18	637.78	21.50
11:06:19	637.78	21.00

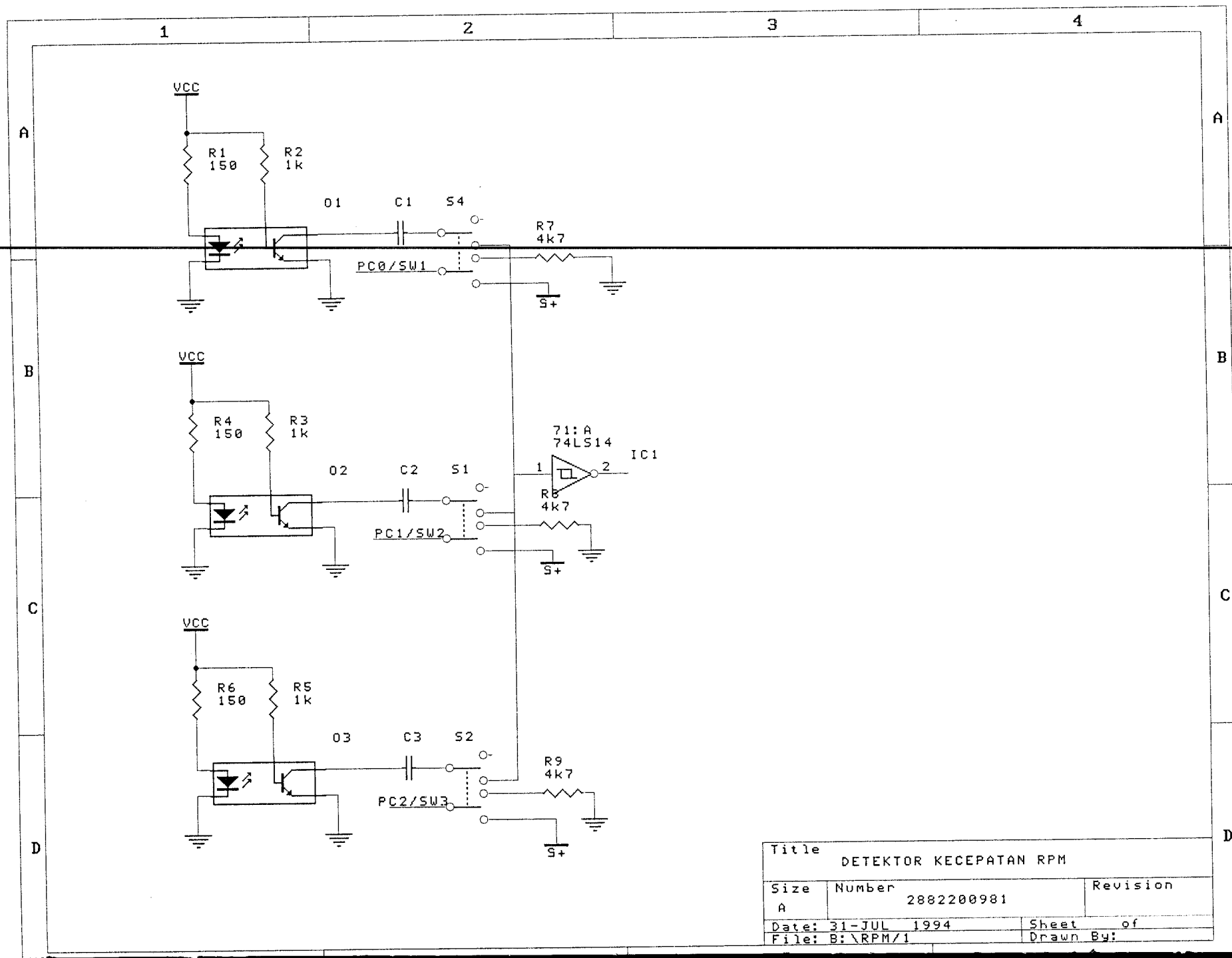


Title		
RANGKAIAN BUFFER		
Size	Number	Revision
A	ERMA SURYANI 288220981	
Date: 31-JUL 1994	Sheet of	
File: B:BUF/1	Drawn By:	



Title		
MEMORI EKSTERNAL		
Size	Number	Revision
A	ERMA SURYANI 2082200981	
Date: 31-JUL 1994	Sheet	of
File: B:\MEMO/1	Drawn By:	

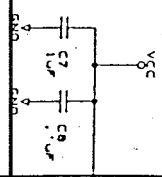




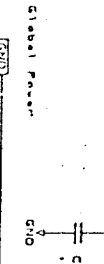
Title			DETEKTOR KECEPATAN RPM
Size	Number	Revision	
A	2882200981		
Date:	31-JUL 1994	Sheet	of
File:	B:\RPM/1	Drawn By:	

MCU12-321

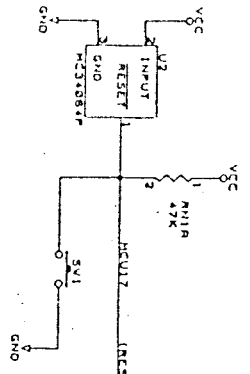
EEPROM



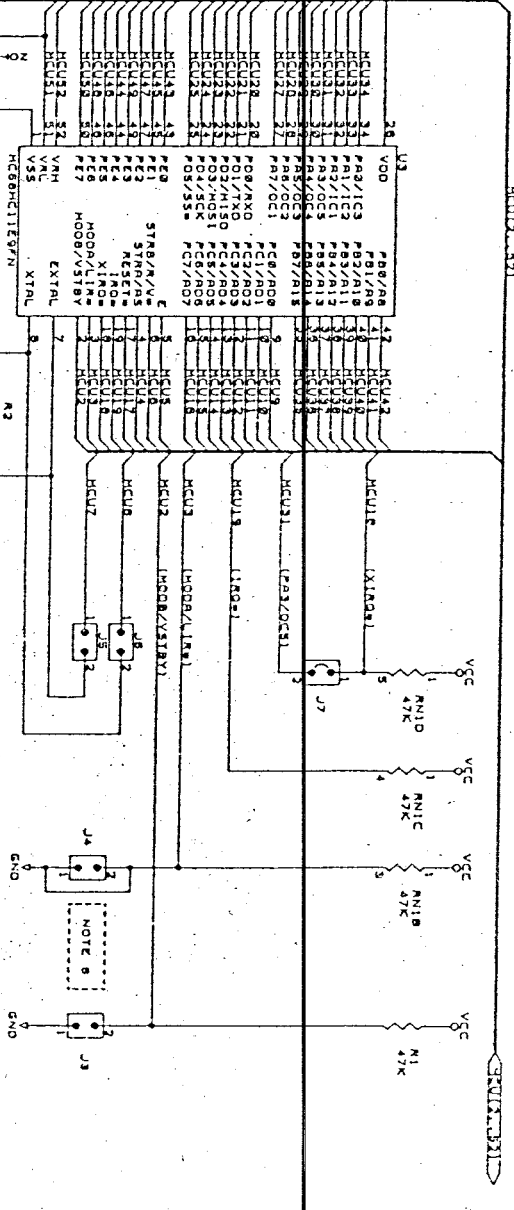
BUFFALO MONITOR



MASTER RESET

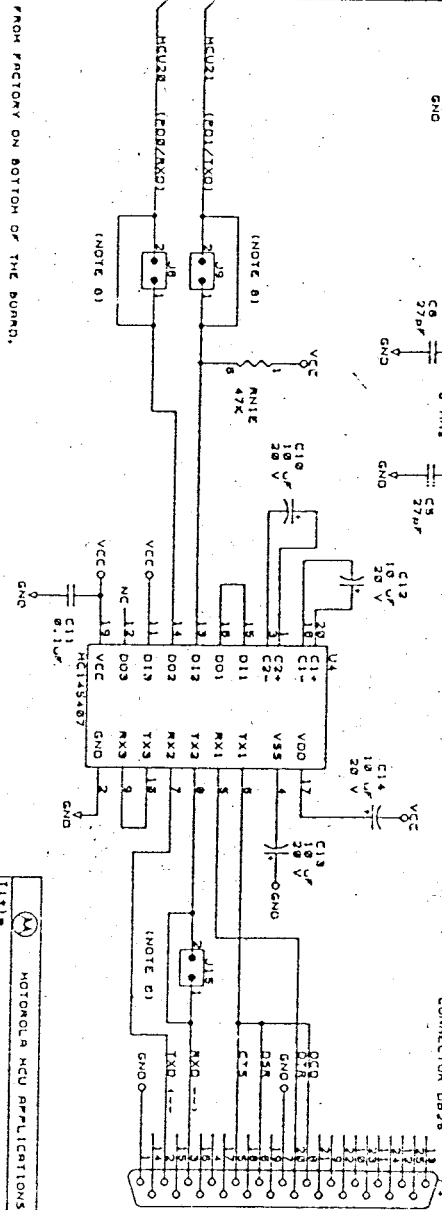


NOTE 8: DEFAULT CUT TRACES INSTALLED FROM FACTORY ON BOTTOM OF THE BOARD.
NOTE 9: X1 IS SHIPPED AS A CERAMIC RESONATOR WITH BUILT IN CAPACITORS.
HOLES ARE PROVIDED FOR A CRYSTAL AND TWO CAPACITORS.

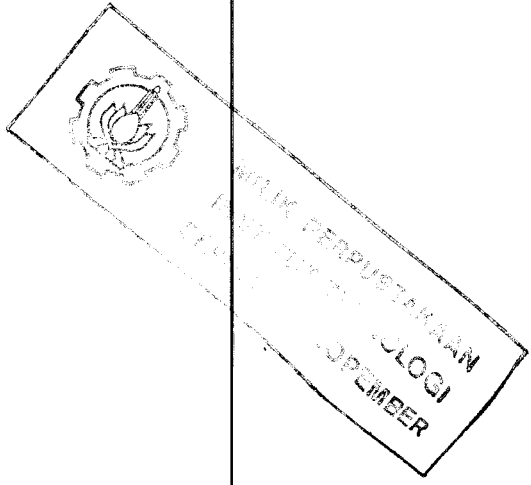
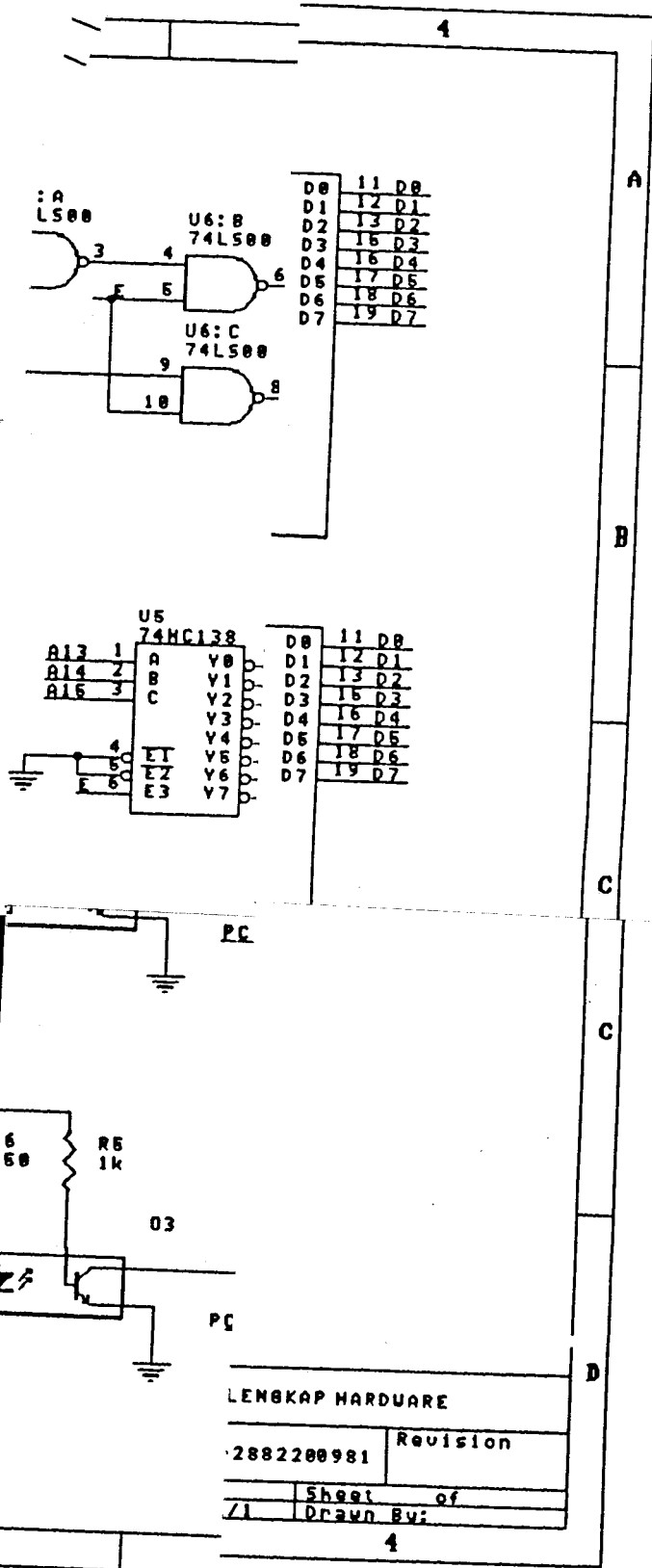


User's Terminal or PC

CONNECTOR DB25



11111	MOTOROLA MCU APPLICATIONS
11111	MCU12-321 EVU
11111	Size Document Number
11111	830VE9319AV
11111	Date: 22-1992
11111	2 of 2



LENGKAP HARDWARE	
2882200981	Revision
Sheet of	
1	Drawn By:


```

1 A      1000      REGBASE equ    $1000
2 A      0039      OPTION equ    $39
3 A      0010      TIC1 equ    $10
4 A      0012      TIC2 equ    $12
5 A      0024      TMSK2 equ    $24
6 A      0035      BPROT equ    $35
7 A      0020      TCTL1 equ    $20
8 A      0023      TFLG1 equ    $23
9 A      0021      TCTL2 equ    $21
10 A     102E      SCSR equ    $102E
11 A     102F      SCDR equ    $102F
12 A     102C      SCCR1 equ    $102C
13 A     102D      SCCR2 equ    $102D
14 A     102B      BAUD equ    $102B
15 A 0000
16 A
17 A      $-----
18 A      $ alamat untuk PPI 8255 dan LED
19 A      $-----
19 A     0200      portA equ    $200
20 A     0201      portB equ    $201
21 A     0202      portC equ    $202
22 A     0203      portCW equ    $203
23 A 0000
24 A      *****
25 A      $ Isi Memori
26 A      *****
27 A     6000      KMH equ    $6000
28 A     8000      RPM equ    $8000
29 A     4400      KMH_PTR equ    $4400
30 A     4402      RPM_PTR equ    $4402
31 A 0000
32 A      *****
33 A      $ REVERSE MEMORY BYTE
34 A      *****
35 A     4404      frste1 equ    $4404
36 A     4406      perc1 equ    $4406
37 A     4408      temp1 equ    $4408
38 A     440A      freqh1 equ    $440A
39 A     440C      frste2 equ    $440C
40 A     440E      perc2 equ    $440E
41 A     4410      temp2 equ    $4410
42 A     4412      freqh2 equ    $4412
43 A 0000
44 A     4413      out1 equ    $4413
45 A 0000
46 A     4415      in equ    $4415
47 A     4416      out equ    $4416
48 A 0000
49 A     2000      org    $2000

```

50 A

```
51 A          #####
52 A          * Inisialisasi SCI
53 A          #####
54 A 2000 36      onsci      psha
55 A 2001 8630          ldaa      #$30
56 A 2003 B7102B          staa      BAUD
57 A 2006 8600          ldaa      #$00
58 A 2008 B7102C          staa      SCCR1
```

M68HC11 Absolute Assembler Version 2.70c:\revws\ta.ASC

```
59 A 200B 860C          ldaa      #$0C
60 A 200D B7102D          staa      SCCR2
61 A 2010 32          pula
62 A 2011 39          rts
63 A
64 A 2012 3C      kirim      pshx
65 A 2013 36          psha
66 A 2014 37          pshb
67 A 2015 F6102E      lp      ldab      SCSR
68 A 2018 C420          andb      #$80
69 A 201A 27F9          beq      lp
70 A 201C F64416          ldab      out
71 A 201F F7102F          stab      SCDR
72 A 2022 33          pulb
73 A 2023 32          pula
74 A 2024 38          pulx
75 A 2025 39          rts
76 A
77 A
78 A
79 A
80 A 2026 183C      delay1s      pshy
81 A 2028 3C          pshx
82 A 2029 CE0019          ldx      #25
83 A 202C 18CE2CA3      loop1      ldy      #11427      ; 34250
84 A 2030 1809      loop2      dey
85 A 2032 26FC          bne      loop2
86 A 2034 09          dex
87 A 2035 26F5          bne      loop1
88 A 2037 38          pulx
89 A 2038 1838          puly
90 A 203A 39          rts
91 A
92 A      203B      delay      equ      *
93 A 203B 3C          pshx
94 A 203C 183C          pshy
95 A 203E CE00FF          ldx      #$FF
96 A 2041 18CE0150      wait      ldy      #150
```

```

97 A 2045 1809      loopy    dey
98 A 2047 26FC              bne      loopy
99 A 2049 09          dex
100 A 204A 26F5              bne      wait
101 A 204C 1838              puly
102 A 204E 38           pulx
103 A 204F 39           rts
104 A
105 A      2050      delay2    equ      *
106 A 2050 3C              pshx
107 A 2051 183C              pshy
108 A 2053 CE0100           ldx      #100
109 A 2056 18CE0200        wait2     ldy      #200
110 A 205A 1809      loopy2     dey
111 A 205C 26FC              bne      loopy2
112 A 205E 09          dex
113 A 205F 26F5              bne      wait2
114 A 2061 1838              puly
115 A 2063 38           pulx
116 A 2064 39           rts

```

M68HC11 Absolute Assembler Version 2.70c:\revws\ta.ASC

```

117 A
118 A      *****
119 A      * Main program
120 A      *****
121 A 2065 8E5FFF        main     lds      #5fff
122 A 2068 BD2000              jsr      onsci      ; init SCI
123 A
124 A 206B 86B9              ldaa     #89          ; control word PPI 8255
125 A 206D B70203           staa     portCW      ; PA in, PB out, PC inp
126 A 2070 CE6000           ldx      #KMH          ; #6000
127 A 2073 FF4400           stx      KMH_PTR      ; #4000
128 A 2076 CE8000           ldx      #RPM          ; #8000
129 A 2079 FF4402           stx      RPM_PTR      ; #4002
130 A
131 A      *****
132 A      * Awal Program
133 A      *****
134 A 207C CE1000        pertop    ldx      #REGBASE
135 A 207F 8604              ldaa     #X00000100      ; ic2
136 A 2081 A721              staa     TCTL2,x      ; edgel:edgela = 0:1, risingedge
137 A 2083 8602              ldaa     #02
138 A 2085 A723              staa     TFLG1,x      ; clear any old orl flag
139 A 2087 1F2302FC        brclr    TFLG1,x,#02 *      ; loop here until edge
140 A 208B EC12              ldd      TIC2,x      ; read time of first edge
141 A 208D FD440C           std      frste2      ; save first capture value
142 A
143 A 2090 8602              ldaa     #02
144 A 2092 A723              staa     TFLG1,x      ; clear iclf before next edge

```

```

145 A 2094 1F2302FC      brclr    TFLB1,x,#02 #      ; loop here until edge
146 A 2098 EC12          ldd      TIC2,X      ; read time of second edge
147 A 209A B3440C        subd     frste2      ; 2nd-1st --> d
148 A 209D FD440E        std      perc2      ; save result (period in cys)
149 A 20A0 FE440E        ldx      perc2      ; periode in cycles
150 A 20A3 CC0020      okp2  ldd      #32      ; x=periode; d=32
151 A 20A6 03           fdiv     #32      ; d/x --> x; r-->d
152 A 20A7 FF4410        stx      temp2
153 A                    *****
154 A                    # display frekwensi
155 A                    *****
156 A 20AA FC4410        ldd      temp2      ; (2**20)f; where f=freq+(10**6)
157 A 20AD 04           lsr     #16
158 A 20AE 04           lsr     #16
159 A 20AF 04           lsr     #16      ; A=(2**9)f; D=(2**17)f
160 A 20B0 7F4412        clr      freqh2     ; clear upper half of hex freq loc
161 A 20B3 B74413        staa     freqh2+1    ; freqh is a temp = 512f
162 A 20B6 04           lsr     #16
163 A 20B7 04           lsr     #16      ; now d=(2**15)f or 32,768f
164 A 20B8 FD4410        std      temp2      ; needs to be in mem for subtract
165 A 20BB 8F           xgdx     #16
166 A 20BC B34410        subd     temp2      ; d=(2**20)f; x=(2**15)f
167 A 20BF 8F           xgdx     #16      ; 1st subtraction
168 A 20C0 04           lsr     #16      ; working result --> x; d=(2**15)f
169 A 20C1 FD4410        std      temp2      ; A=(2**6)f; D=(2**14)f
170 A 20C4 BB4413        adda     freqh2+1    ; put in mem so we can subtract
171 A 20C7 B74413        staa     freqh2+1    ; (512*64)f
172 A 20CA 8F           xgdx     #16      ; update low half of freqh
173 A 20CB B34410        subd     temp2      ; D=1,015,808*f; x=junk
174 A 20CE F34412        addd     freqh2     ; 999,424*f
                                           ; 1,000,000*f = frequency

```

M68HC11 Absolute Assembler Version 2.70c:\revws\ta.ASC

```

175 A 20D1 FD4412        std      freqh2     ; save the 16-bit binary result
176 A
177 A 20D4 FC4412        ldd      freqh2
178 A 20D7 1A83005A      cpd      #5A      ; 90hz
179 A 20DB 2D0E          blt      diskmh
180 A 20DD 5F           clrb
181 A 20DE C601          ldab     #01
182 A 20E0 F70201        stab     portB
183 A 20E3 BD203B        jsr      delay
184 A 20E6 C600          ldab     #00
185 A 20E8 F70201        stab     portB
186 A
187 A
188 A                    *****
189 A                    # display dalam hertz (km/h)
190 A                    *****
191 A 20EB BD203B        diskmh  jsr      delay
192 A 20EE FE4400        ldx      KMH_PTR

```

```

193 A 20F1 FC4412      ldd      freqh2
194 A 20F4 E700        stab     0,x
195 A 20F6 08          inx
196 A 20F7 2900        blt      d_rpm
197 A
198 A                  *****
199 A                  * rising edge pertama siap dideteksi
200 A                  *****
201 A 20F9 CE1000      d_rpm    ldx      #REGBASE
202 A 20FC 8610        ldaa     #X00010000
203 A 20FE A721        staa     TCTL2,x
204 A 2100 8604        ldaa     #*04
205 A 2102 A723        staa     TFLG1,x          ; clear any old ocl flag
206 A 2104 1F2304FC    brclr   TFLG1,x,#*04 *      ; loop here until edge
207 A 2108 EC10        ldd      TIC1,x          ; read time of first edge
208 A 210A FD4404      std      frstet1          ; save first capture value
209 A
210 A 210D 8604        ldaa     #*04
211 A 210F A723        staa     TFLG1,x          ; clear iclf before next edge
212 A
213 A 2111 1F2304FC    brclr   TFLG1,x,#*04 *      ; loop here until edge
214 A 2115 EC10        ldd      TIC1,x          ; read time of second edge
215 A 2117 B34404      subd     frstet1          ; 2nd- 1st --> d
216 A 211A FD4406      std      percl          ; save result (period in cycs)
217 A 211D FE4406      ldx      percl          ; periode in cycles
218 A 2120 CC0020      okpl     ldd      #32          ; x=periode; d=32
219 A 2123 03          fdiv          ; d/x --> x; r-->d
220 A 2124 FF4408      stx      tempi
221 A
222 A                  *****
223 A                  * display frekwensi
224 A                  *****
225 A 2127 FC4408      ldd      tempi          ; (2**10)f; where f=freq+(10**6)
226 A 212A 04          lsr      d
227 A 212B 04          lsr      d
228 A 212C 04          lsr      d          ; A=(2**9)f; D=(2**17)f
229 A 212D 7F440A      clr      freqh1          ; clear upper half of hex freq loc
230 A 2130 B74408      staa     freqh1+1          ; freq is a temp = 512f
231 A 2133 04          lsr      d
232 A 2134 04          lsr      d          ; now d=(2**15)f or 32,768f

```

M68HC11 Absolute Assembler Version 2.70c:\revws\ta.ASC

```

233 A 2135 FD4408      std      tempi          ; needs to be in mem for subtract
234 A 2138 8F          xgdx          ; d=(2**20)f; x=(2**15)f
235 A 2139 B34408      subd     tempi          ; 1st subtraction
236 A 213C 8F          xgdx          ; working result --> x; d=(2**15)f
237 A 213D 04          lsr      d          ; A=(2**6)f; D=(2**14)f
238 A 213E FD4408      std      tempi          ; put in mem so we can subtract
239 A 2141 B84408      adda     freqh1+1          ; (512**64)f
240 A 2144 B74408      staa     freqh1+1          ; update low half of freqh

```

```

241 A 2147 8F          xgdx          ; D=1,0 5,808*f; x=junk
242 A 2148 B34408      subd      temp1      ; 999,424*f
243 A 214B F3440A      addd      freqhi    ; 1,000,000*f = frequency
244 A 214E FD440A      std        freqhi    ; save the 16-bit binary result
245 A
246 A 2151 B60202      ldaa      portC
247 A 2154 8401        anda      #$01
248 A 2156 8101        cmpa      #$01
249 A 2158 2614        bne       gigi2
250 A 215A FC440A      ldd        freqhi
251 A 215D 1A830032    cpd        #$32
252 A 2161 2D42        blt        dis_rpm
253 A 2163 8601        ldaa      #$01
254 A 2165 B70200      staa      portA
255 A 2168 BD203B      jsr        delay
256 A 216B 7E21A5      jmp        dis_rpm
257 A
258 A 216E B60202      gigi2    ldaa      portC
259 A 2171 8402        anda      #$02
260 A 2173 8102        cmpa      #$02
261 A 2175 2614        bne       gigi3
262 A 2177 FC440A      ldd        freqhi
263 A 217A 1A830041    cpd        #$41
264 A 217E 2D25        blt        dis_rpm
265 A 2180 8602        ldaa      #$02
266 A 2182 B70200      staa      portA
267 A 2185 BD203B      jsr        delay
268 A 2188 7E21A5      jmp        dis_rpm
269 A
270 A 218B B60202      gigi3    ldaa      portC
271 A 218E 8404        anda      #$04
272 A 2190 8104        cmpa      #$04
273 A 2192 2611        bne       dis_rpm
274 A 2194 FC440A      ldd        freqhi
275 A 2197 1A830050    cpd        #$50
276 A 219B 2D08        blt        dis_rpm
277 A 219D 8604        ldaa      #$04
278 A 219F B70200      staa      portA
279 A 21A2 BD203B      jsr        delay
280 A
281 A
282 A
283 A          *****
                $ Simpan di RAM
284 A          *****
285 A 21A5 FE4402      dis_rpm  ldx        RPM_PTR
286 A 21A8 FC440A      ldd        freqhi
287 A 21AB E700        stab      0,x
288 A 21AD 08          inx
289 A
290 A

```

```

291 A          *****
292 A          * Cek tombol pc7 apakah ditekan
293 A          *****
294 A
295 A 21AE B60202      cek      ldaa      portC
296 A 21B1
297 A 21B1 8480          anda      #$80
298 A 21B3 8100          cmpa      #$00
299 A 21B5 2636          bne       pertop1
300 A
301 A 21B7 CE6000          ldx      #$6000
302 A 21BA 18CE8000        ldy      #$8000
303 A
304 A 21BE BD2012      lagi      jsr      kirim
305 A 21C1 BD203B          jsr      delay
306 A 21C4 E600          ldab     0,x
307 A 21C6 F74416        stab     out
308 A 21C9 BD2012          jsr      kirim
309 A 21CC BD203B          jsr      delay
310 A 21CF 08           inx
311 A 21D0 BD2050          jsr      delay2
312 A 21D3 BD2012          jsr      kirim
313 A 21D6 BD203B          jsr      delay
314 A 21D9 18E600        ldab     0,y
315 A 21DC F74416        stab     out
316 A 21DF BD2012          jsr      kirim
317 A 21E2 BD203B          jsr      delay
318 A 21E5 1808          iny
319 A
320 A 21E7 188C9FFF        cpy      #$ffff
321 A 21EB 2DD1          blt       lagi
322 A
323 A 21ED 7E207C        pertop1  jmp      pertop
324 A                      end

```

SYMBOL TABLE: Total Entries= 57

BAUD	1028	gigi2	216E
BPROT	0035	gigi3	218B
KMH	6000	in	4415
KMH_PTR	4400	kirim	2012
OPTION	0039	lagi	21BE
REGBASE	1000	loop1	202C
RPM	8000	loop2	2030
RPM_PTR	4402	loopy	2045
SECR1	102C	loopy2	205A
SECR2	102D	lp	2015
SECDR	102F	main	2065
SECSR	102E	okp1	2120
TCTL1	0020	okp2	20A3

TCTL2	0021	onsci	2000
TFLG1	0023	out	4416
TIC1	0010	out1	4413
TIC2	0012	perc1	4406
TNSK2	0024	perc2	440E
cek	21AE	pertop	207C
d_rpm	20F9	pertopl	21ED
delay	203B	portA	0200
delay1s	2026	portB	0201
delay2	2050	portC	0202
dis_rpm	21A5	portCW	0203
diskmh	20EB	temp1	4408
freg1	440A	temp2	4410
freg2	4412	wait	2041
frste1	4404	wait2	2056
frste2	440C		

Total errors: 0

05 MAY 1993

EE 1799 TUGAS AKHIR 6 SKS

Nama Mahasiswa : Erma Suryani
NRP : 2882200981
Bidang Studi : Elektronika
Dosen Wali : Ir. Nawanto Wibowo
Dosen pembimbing : 1. Ir. Nawanto Wibowo
2. Ir. Tasripan
Judul Tugas Akhir : Kotak Hitam untuk Kendaraan Umum
dengan Menggunakan Mikrokontroler 68HC11E9.

Uraian Tugas Akhir :

Untuk memonitor kelakuan pengendara kendaraan umum selama di perjalanan terutama pada bis, maka penting untuk dibuat suatu alat yang dapat merekam semua kejadian berupa data kecepatan kendaraan dan kecepatan roda gigi terhadap waktu.

Selain pengambilan data, alat ini juga dapat memberikan peringatan jika kecepatannya melebihi batas yang ditentukan. Data disimpan dalam RAM, dan setelah perjalanan usai, data di RAM diambil untuk disimpan dalam bentuk file dan akhirnya ditampilkan dalam bentuk grafik.

Dari grafik ini pemilik kendaraan dapat mengetahui apakah pengemudi ugal-ugalan atau sering melakukan pengereman mendadak. Hal ini terlihat jika grafik kecepatan turun dalam sesaat.

Surabaya, 29 April 1993

Menyetujui

Dosen Pembimbing I

Dosen Pembimbing II



(Ir. Nawanto Wibowo)



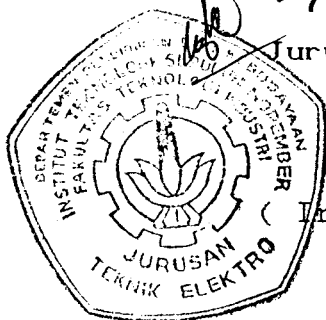
(Ir. Tasripan)

Menyetujui
Bidang Studi Elektronika
Koordinator

30/93
14

Ir. Soetikno)

Mengetahui
Jurusan Teknik Elektro FTI ITS
Ketua



(Ir. Katjuk Astrowulan, MSEE)
NIP. 130 687 438

USULAN TUGAS AKHIR

- A. JUDUL : Kotak Hitam untuk Kendaraan Umum dengan Menggunakan Mikrokontroler 68HC11E9.
- B. RUANG LINGKUP : - Instrumentasi Elektronika.
- Teknik Digital.
- Bahasa Pemrograman.
- Pengukuran Elektronika.
- Rangkaian Linear.
- Mikroelektronika.
- C. LATAR BELAKANG : Adanya keinginan untuk mengetahui kelakuan pengendara kendaraan umum selama di perjalanan terutama pada bis dan sekaligus memberi peringatan jika kecepatannya melebihi batas-batas yang ditentukan, maka perlu dibuat suatu alat yang dapat merekam semua kejadian selama di perjalanan. Kejadian ini direkam dalam bentuk data yang meliputi kecepatan kendaraan dan kecepatan putaran roda gigi terhadap waktu. Selain pengambilan data juga diperlukan adanya indikasi apakah si pengendara berada di luar atau di dalam kota, untuk mengontrol kecepatannya.

D. PENELAAHAN STUDI : Perencanaan dan pembuatan alat yang dapat mengambil data selama dalam perjalanan dan memberinya peringatan jika kecepatan kendaraan melebihi batas-batas yang ditentukan. Jika terjadi hal-hal yang tidak diinginkan maka alat ini dapat dijadikan informasi yang akurat pada saat berlangsungnya kejadian. Data disimpan di RAM, dan jika perjalanan telah berakhir data yang terdapat di RAM ini disimpan dalam bentuk file agar dapat ditampilkan dalam bentuk grafik. Untuk menyimpan data dan mengontrol kecepatan ini maka digunakan mikrokontroler 68HC11E9.

E. TUJUAN : Pembuatan alat yang dapat mengambil dan menyimpan data kendaraan umum selama di perjalanan.

F. LANGKAH : - Survey Lapangan.
- Studi Literatur.
- Perencanaan Rangkaian.
- Penyediaan Komponen dan Peralatan.
- Pengukuran Peralatan dan percobaan.
- Kesimpulan Hasil Percobaan.

G. RELEVANSI : Peralatan dirancang sedemikian rupa sehingga dapat diandalkan untuk perjalanan jauh dan merupakan kontrol bagi pengendara agar

lebih berhati-hati jika menjalankan kendaraan, sehingga akan lebih menjamin keselamatan penumpang dan bagi si pemilik kendaraan dapat menghemat biaya operasional karena peralatannya tidak akan cepat aus. Jadi dalam hal ini terdapat keuntungan di semua pihak.

H. JADWAL KEGIATAN :

No	Uraian Tugas	Bulan ke					
		I	II	III	IV	V	VI
1.	Survey Lapangan	■■■■■					
2.	Studi Literatur	■■■■■■■■■■					
3.	Perencanaan Rangkaian		■■■■■■■■■■				
4.	Pembuatan Alat			■■■■■■■■■■			
5.	Pengukuran/Perco- baan			■■■■■■■■■■■■■■■■■■■■			
6.	Kesimpulan				■■■■■■■■■■		